web designer™

HTML5 | **CSS3** | **jQuery** | **WordPress**

www.webdesignermag.co.uk

RESPONSIVE DESIGN
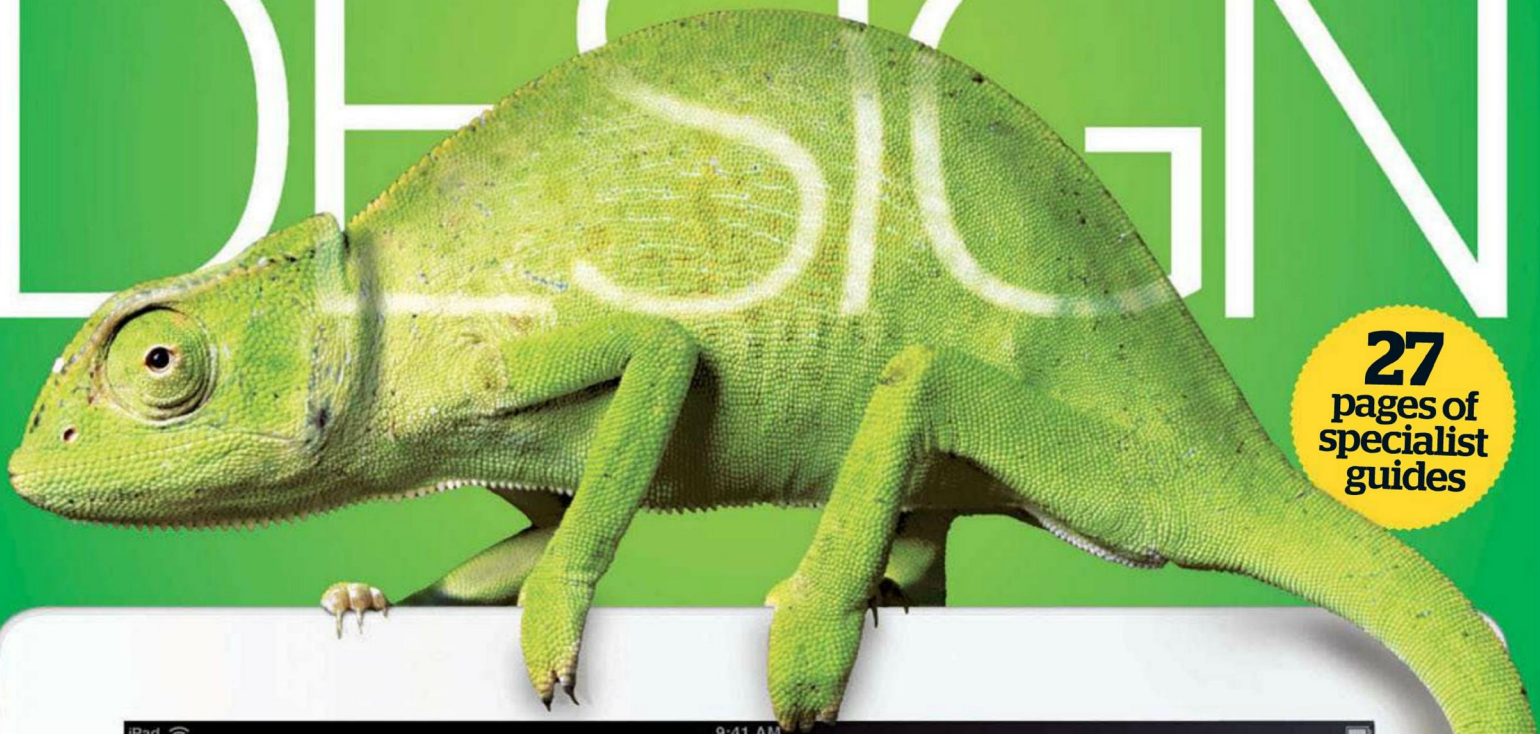
Build adaptive websites for every environment spanning desktop, smartphone and tablet

HOW TO

Apply fluid layouts with Masonry • Responsify your WordPress templates
Create a responsive slider • Optimise text and images with jQuery plug-ins

iPad 🛜   9:41 AM

SARAH PARMENTER
Star designer gives her view on the challenges posed by adaptive web design

MODERNIZR 2.5
Next-gen browser detection with this JavaScript library

Welcome to the issue

Mark Billen

Highlight

❝Listening to, monitoring and engaging with social media is becoming more and more fundamental❞

Luke Townsend, product director at Found, describes how social networks are crucial to driving web traffic. **Page 30**

The time for designing more responsively is finally upon us!

It's pretty fair to say that web design is a discipline that doesn't like to make things easy for itself. Remember the bad old days when we used to have to include countless workarounds within our page designs, just to cater for the compatibility quirks between browsers? Well, although we have finally reached a point where HTML5 is broadly supported by IE, Firefox, Safari and Chrome, it seems life hasn't gotten any easier! These days the waters are muddied by hardware and a whole new ecosystem of mobile browsers, screen sizes and ways to interact. Smartphones, tablets and even consoles have given more choice as to how we consume web content and developers face a growing challenge in how to target them. Naturally, the idea of building multiple versions of the same site is not something most of us would relish. So this is where the notion of a more 'responsive' design ideology appears, whereby content adapts to its environment. Ethan Marcotte, in an article for *A List Apart*, acknowledged fluid grids, flexible images and CSS media queries as the three 'technical ingredients' for responsive web design; and it's these key methods that make the concept work. So this month we wanted to delve deeper into a subject that is arguably causing the biggest buzz in the community today, with a whopping 27 pages of coverage. Spanning the lead feature (from page 38), a developer look at Modernizr 2.5, three themed tutorials and more, you'll not only get a definitive idea of what responsive design really means, but also all the practical tricks you need to implement it.

Mark

❝These days the waters are muddied by hardware and a whole new ecosystem of mobile browsers, screen sizes and ways to interact❞

Excitographic

Plotting the features that got us in a frenzy over the month...

- Steve Jenkins, Features Editor
- Mark Billen, Editor
- Steven Mumby, Designer
- Ben Martin, Sub Editor

"With plenty to hide, the team seem less enthusiastic by Google's new policy on handling user data"

"The initial excitement for haptic technology subsides when we finally realise its only got a retina display...boo!"

Radar chart labels: FlashJS, Responsive design, Adobe Creative Cloud, Samsung Galaxy S3, WebRTC in Opera Mobile 12, Zynga.com, Windows 8, New Apple iPad, Modernizr 2.5, Google's Privacy Policy, ImagineSubs 2.0
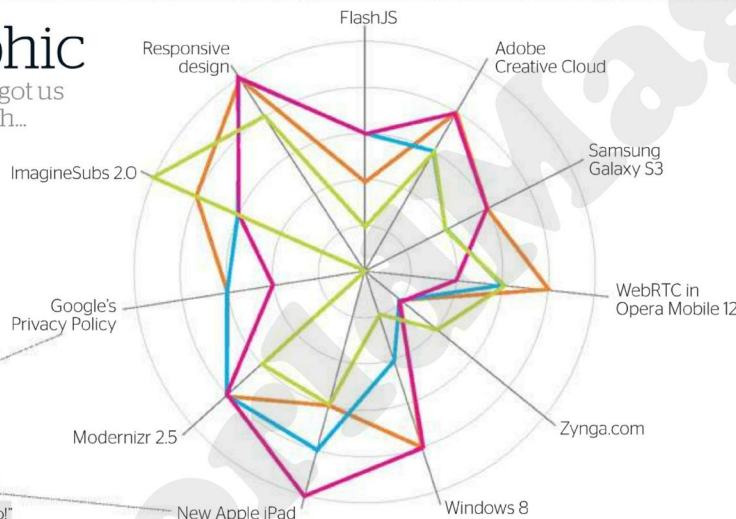
Contact us to join in the conversation:

Join us on Facebook:
facebook.com/webdesignermag

Tweet us:
@webdesignermag

Email the team:
webdesigner@imagine-publishing.co.uk

Visit the website:
www.webdesignermag.co.uk

Turn over to the contents to discover what's going to get you excited this issue...

This issue's panel of experts

Welcome to that bit of the mag where we learn more about the featured writers and contributors...

Kayla Knight

Kayla is a freelance web designer and developer with a primary focus on WordPress and blog design. She also loves blogging herself, and manages a web-design agency (42function.com) that helps small businesses and startups grow. She joins us this month to talk responsive design; you can check out her portfolio at www.kaylaknight.com. **Page 38**

❝ **She also loves blogging herself, and manages a web-design agency** ❞

Mark Shufflebottom

Mark is an Adobe Education Leader and programme co-ordinator of BA (Hons) Interactive Media Production at Bournemouth University. This issue Mark concerns himself with location-aware sites and the Geolocation API. **Page 50**

Steve Jenkins

Steve is Features Editor on **Web Designer** and has been creating websites since 1999. He's interested most in HTML, CSS and WordPress but this month veers towards jQuery to look at a cool way to add in-line Twitter feeds. **Page 60**

Louis Simoneau

Louis slings Ruby on Rails for Flippa, and hosts the SitePoint podcast for designers and developers. He makes his **Web Designer** debut this issue by picking apart the new features of the Modernizr 2.5 framework. **Page 76**

Thord Daniel Hedengren

Known as TDH for short, Thord is an author and freelance journalist, as well as a huge supporter of open-source software. When he's not playing with words he runs the Odd Alice web agency in Stockholm, Sweden. **Page 54**

Sam Hampton-Smith

A keen supporter of web standards, Sam loves getting the most out of HTML and CSS. When not writing for **Web Designer**, he runs a graphic design agency in Scotland and you can follow his antics on Twitter via the username @samhs. **Page 46**

Matt Gifford

Matt is a lead RIA consultant developer who specialises in ColdFusion, Flash and AIR development. With over ten years' industry experience across various sectors, he is a regular presenter at national and international conferences. **Page 82**

Eric Freeman

Alongside Elisabeth Robson, Eric is co-author of the books *Head First HTML & CSS* and *Head First HTML5 Programming*. A computer scientist and former CTO of Disney.com, he's back to offer a first lesson in using Web Workers. **Page 86**

Elisabeth Robson

Elisabeth is the co-founder of the online education experience WickedlySmart, providing various training on web technologies. Joining forces with Eric, she's here to collaborate on part one of our Web Workers guide. **Page 86**

---

contents

Cutting-edge features, techniques and inspiration for web creatives

Chat with the team and other readers and discuss the latest tech, trends and techniques. Here's how to stay in touch...

webdesigner@imagine-publishing.co.uk · @WebDesignerMag · www.webdesignermag.co.uk

Inside...

<header>

Discussing the hottest topics from the web-design world

Cover focus

RESPONSIVE DESIGN

38
All you need to know about adding great adaptive design techniques to your web skillset
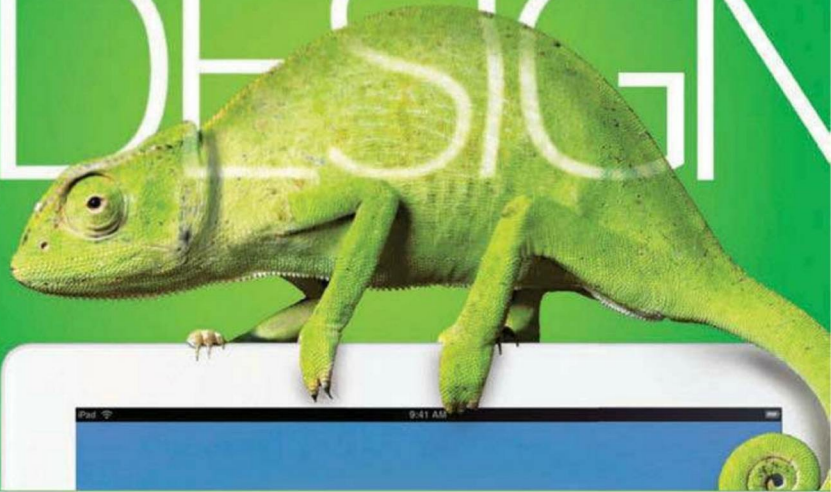
30 Pro file: Found
SEO is still big business for this London agency

96 Trend map: Helsinki
All the hotspots of this creative European city

08 Adobe casts mobile Shadow
Examining the new site-testing tool

26 Design diary: demodern
Extreme creativity for Nike's Chosen Series site

On the free CD

A premium collection of designer resources worth over £100!

- 110 minutes of video tutorials
- 2 responsive WordPress themes
- 20 Stitchy circles vector pack
- 15 repeating tile textures

112

Web workshop
Re-creating the web's hottest styles

<tutorials>

Web gurus take you step-by-step through professional techniques

web {developer;

NEW! Dedicated 15-page section offering features and tutorials for coders
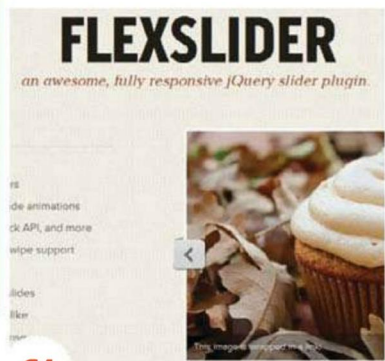
Never miss an issue **Subscribe** Turn to page 100 now. USA readers turn to page 108 for an exclusive offer

‹header›

Discussing the hottest topics in the web-design world

If you have a creative project, new web product or great designer story, contact the editorial desk

webdesigner@imagine-publishing.co.uk @WebDesignerMag
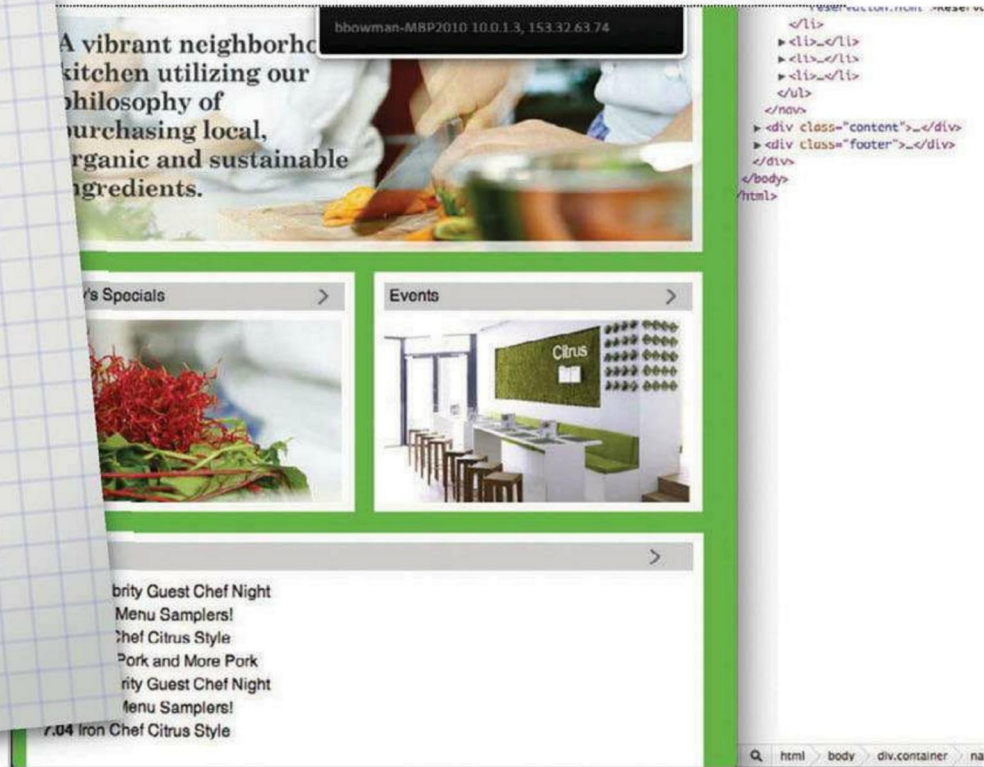
Adobe casts a new Shadow

Testing your website designs for multiple mobile devices has just been made a whole lot easier thanks to this clever new program

The web is no longer a desktop-centric event. Smartphones and tablets are now such an integral part of the online experience that designers and developers cannot simply sit back and admire their desktop creations. Android and iOS platforms power millions of tablets and smartphones offering a new version of the web; this, of course, means it is no longer simply a matter of testing on the major browsers across the popular resolutions. Mobile devices add a further collection of resolutions with the added

design bonus of accommodating for both portrait and landscape orientations. Testing on all platforms is more critical, and complicated, than it has ever been.

Adobe and the web-design community have long been the best of friends and it's not without good reason. Its latest addition to the toolset of digital creatives is Adobe Shadow.

In a nutshell, Shadow is a tool for testing the pages you have created to ascertain their compatibility with multiple mobile devices. Adobe has described the new piece of multi-device

testing software 'as a new inspection and preview tool that allows front-end web developers and designers to work faster and more efficiently by streamlining the preview process, making it easier to customise websites for mobile devices.'

To ensure that users are receiving the best output possible, Adobe has stated: 'Shadow will be updated regularly to stay ahead of web standards, web browser updates and support for new mobile devices entering the market'.

> **Shadow is undoubtedly a great tool, and like all the best ideas, it's a simple one**

The new software enables creatives to sync up wirelessly with a host of devices. All the user needs to do is add a passcode and get connected to the computer where the actual page development is taking place. Users can add multiple devices peripherally to the main unit, and when a page is updated all devices will update automatically. This gives designers and developers an instant picture of exactly how the current page is going to look on the connected gadget. Emulators can be excellent solutions but are often limited in scope and operation, and it goes without saying there's no real substitute when testing pages than to trial a page on the device it's going to be seen on.

Shadow is undoubtedly a great tool, and like all the best ideas, it's a simple one. If you have yet to try Shadow, put it on your to-do list; we promise that you won't regret it.

Adobe Shadow releases

Adobe Shadow is more than a single download; it has an accompanying set of assets that help extend its capabilities. Here we list the essential companions for this new piece of software.

Download Shadow
adobe.ly/xlAc43

Shadow installer for Mac and Windows

Mac download
bit.ly/yABNPh
Windows download
bit.ly/zAlGrW

Shadow client for iOS
bit.ly/AlvVXb

Shadow client for Android
bit.ly/wyLuAA

Google Chrome extension
This allows users to browse the desktop while all connected devices remain in sync.
bit.ly/uJofC8

Shadow on Adobe TV
adobe.ly/wsSlwm

COMMENT

Mark Shufflebottom
Adobe education leader
www.webspaceinvader.com

Adobe has announced a number of new tools for the web-design community over the last year or so that have been met with less than eager anticipation (Muse is one that might jump to mind). So cynics might well be wondering what Shadow is all about. Fortunately Shadow offers a really valuable feature set to front-end designers. There is no doubt that it is incredibly difficult and time consuming to test web designs across multiple devices, so any tool that offers to make that job easier is going to be useful. Shadow makes pairing devices to the desktop really simple and, when one device is connected, there doesn't seem to be much benefit. However, it's Shadow's ability to pair multiple devices and see the same design across them that really brings its usefulness to the forefront. This will speed up the design process immensely as problems can be identified, fixed and tested to make sure everything is laid out well on the multitude of devices our clients and customers use.

Discussing the hottest topics in the web-design world

Photoshop for iPad

Adobe launches its famous photo-editing software on the new iPad

Photoshop and the iPad is surely a winning combination, and at only £6.99/$9.99 it's an app not to be missed. The only caveat is that your device needs to be running iOS5 – but who among us hasn't already updated?

The new app takes a host of core features from Photoshop and squeezes them into a beautifully crafted tablet UI. So what's included in the lineup? All the essential features are here, from layers and selection tools to adjustments and filters. Plus, there is a camera fill feature which uses photos from the iPad camera; the Scribble selection tool, which uses your finger to extract an image; and Refine Edge, which uses your fingertip to capture hard-to-select elements like hair.

Beyond the in-built toolset, owners of the app can use AirPrint for wireless printing of PS files, upload projects to Adobe Creative Cloud and open layered files from Adobe Photoshop Touch in Photoshop CS5.

This is a great acquisition for designers on the move, but as you might expect it has its limitations – the most obvious being a maximum image resolution (1,600 x 1,600 pixels), so it's definitely not a direct replacement for desktop Photoshop. This said, for such a nominal price it is a must-have product for any designer who owns an iPad.

> **66** This is a great acquisition for designers on the move, but it does have its limitations **99**

Worldwide smartphone sales

Android: 51% (75m units)
iOS: 24% (35m units)
Symbian: 12% (17m units)
BlackBerry: 9% (13m units)
Microsoft: 2% (2.8m units)
Other: 2%
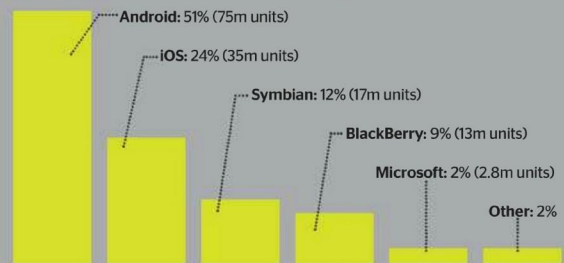
Conclusion

There is no doubt that smartphones and tablets, especially Android and iOS, are the consumer's new best friend. And it is almost inevitable that mobile devices will become more popular than the desktop as a web access point in the not-too-distant future. This means that web designers should be looking to all the necessary tools to ensure the best experience on the frontend whatever the user's platform or device.

‹news cloud›

Bite-sized coverage of the month's trending topics

The first HTML5 document editor
The power of HTML5 is more than just a few semantic tags; it has a vast array of untapped capabilities as demonstrated by the TeamLab Document Editor. The HTML5-based editor is effectively a word processor hosted online. Create, edit and save documents in a typically Word-style interface. Try out the new editor at the TeamLab Demo Portal.

Firefox 11
The beta version of the Mozilla desktop browser is out and ready to test

HTML5
The new iPad
It's finally here with its 2,048 x 1,536 resolution and 4G connection

IE10
The search giant's alternative to iTunes?

Google Play
The Board announces jQuery Foundation to support development of jQuery UI

jQuery

Microsoft unveils IE10 new web standards
The introduction of IE10 and the slow demise of IE6 suggests that Microsoft is joining the new web revolution. A visit to **msdn. microsoft.com/en-gb/IE** gives a comprehensive guide to newly supported HTML5 and CSS3 standards and how to ready your site for the new browser.

Pay As You Go Advertising

Advertise on Google.
Pay only when people click on your ad.

For most types of advertising, you pay for people to **see** your ad. With Google AdWords advertising you only pay when they **click** to visit your website. So you get exactly what you pay for – more customers. And because this is pay-as-you-go advertising, you have complete control.

Control how much you spend
You can set a daily spending limit and an amount you're happy to pay for people to click on your ad. The price is chosen by you, not us, so you never have to worry about going over your budget.

Control who sees your ad
Your ad will only appear when potential customers are searching online for relevant terms you've chosen to describe your business.

Control where and when they see it
You can choose to target your audience by location and time: attract local customers during your opening hours or, if you'd rather, promote yourself to a global audience round the clock.

Voucher missing?

Don't worry, someone in your company may have beaten you to it. Track them down and make sure they're claiming

£50.00* free advertising credit
with Google AdWords for your business.

No minimum term or commitment.

Offer only available to businesses in the UK.

Don't miss out – try it today
We're offering £50.00* free Google AdWords advertising credit for business users**. There's no obligation to continue – you're in control.

To start attracting new customers today:

Visit **google.co.uk/ads/start**

Or call **0800 169 0478*****

Google™

Are you ready for an HD web?

The release of the new iPad with its new high-resolution Retina display raises questions over the web's HD credentials. Web Designer investigates the implications and possible solutions

The great unveiling of the new iPad proved to be in true Apple tradition, promising much but not quite delivering – iPhone 4S, anyone? However, the new version of the Apple tablet does include a couple of key features that may point the way forward for the web: HD screens and 4G connections.

The new Retina display features a 2,048 x 1,536 resolution, 44 per cent greater colour saturation and 3.1 million pixels on a 9.7-inch display. To put this in context this is four times the number of pixels on the iPad 2 and over a million more than HD TV, which has a resolution of 1,920 x 1,080.

Alongside the super-charged Retina display, the new iPad will be offering super-charged mobile broadband speeds too with its inclusion of 4G. Admittedly, UK owners will not be taking advantage of this feature just yet, but hopefully soon (see 'What is 4G?' boxout for more information).

So, what does this mean for us developers and designers? Simply put, it means more work, but this is far too simplistic an answer. Admittedly, designers and developers do not need to do anything; there's no point adapting just because it's there, but the opportunities and solutions are out there waiting.

The obvious enhancements are better-quality imagery and more colours which should lead to better-looking designs. This scenario immediately raises a couple of issues: how do we do it and what about connection speeds? Undoubtedly, better-quality images will incur larger file sizes, which will lead to longer page loading

times. Will 4G help compensate with its souped-up mobile broadband speeds? Of course, but the whole scenario harks back to the bad old days of when Internet Explorer was king and the even more recent advent of mobile sites. What about 3G users? Will they have to wait an eternity to get the high-quality content or will designers/developers have to create separate versions to accommodate the low-end connection? Decisions will need to be made. Do you design for now, the future, or a combination of the two? Whatever the solution plumped for, one thing's for certain: it will not be straightforward.

Better-looking images are just the tip of the iceberg though. Taking a closer look at the HD scenario there are solutions. Mark Bloomfield (**mark.bloomfield.co.za**) suggests the answer is "a bit of CSS3 media query magic; you'd check the resolution of the user's mobile device and, if higher than X, you'd display the higher-res image."

```
001 @media
002 only screen and (-webkit-min-
device-pixel-ratio : 2),
003 only screen and (min-device-
pixel-ratio : 2) {
004 #logo {
005 background: url(images/logo-
iphone.png) 0 0 no-repeat;
background-size: contain;
006  }
007 }
```

A simple solution, but this only brushes the surface. A more in-depth guide,

'Optimising for High Pixel Density Displays', is a must-read that can be found at Menacing Cloud: **bit.ly/aShxvY**.

Of course, it's not only images that can take advantage of better resolutions. Web fonts will be even sharper, which will lead to improved legibility and readability. There is the possibility of standard and HD video being a more immersive viewing experience. But, as already mentioned, how will this impact on download rates? A more gorgeous and engaging web is an experience that users will embrace, but will we have to wait for improved performance to catch up?

The web has been the focus of attention, but it is not the only platform to get a boost from HD. Apps will be greatly affected too, especially on smartphones and tablets; a high-definition screen will introduce both its problems and benefits for app developers.

3G currently restricts app downloads to 20MB. This may change with 4G, but designing for a higher resolution will create bigger apps and go some way to negate 3G connections. On the positive side: games can get ever-more realistic; medical applications, such as X-rays, can be viewed as originally intended; while the addition of a better camera means even better document scanning.

Going back to the question – are we ready for an HD web – the answer is not as simple as yes or no. The technology is ready and waiting, but it is now down to all designers and developers to catch up and start creating solutions.

Brad Frost
Mobile web strategist
& designer
bradfrostweb.com

COMMENT

"The new iPad's Retina screen is definitely going to cause a lot of problems for web designers as much larger images will need serving up in order to look crisp on the new display. Just to put it in perspective, one iOS icon is now larger than the entire original Mac screen, and while native apps can handle this increased size upfront when the app gets downloaded, websites serve these images only when requested. Network speeds haven't evolved fast enough to keep up, so users will feel the pain as they try to get their content.

"What makes matters worse is that conditionally replacing the images is challenging. Jason Grigsby broke down what Apple is doing to serve up hi-res imagery to the Retina display… it's not promising. Apple is serving up the normal-resolution image first, then detecting the hi-res display and loading in the hi-res image. So not only does the user have to download massive images, but they also have to download the lo-res imagery. And keep in mind the user could well be on a 3G data plan with a capped data limit.

"So that's the bad news and I'm not sure there's an easy solution out there. However, for other images like backgrounds, UI elements and icons, there are a lot of techniques that can alleviate problems caused by the Retina screen. We can use CSS3 techniques (gradients, rounded corners, etc), smartly apply media queries, utilise SVG and more. I break these techniques down in a post (**bit.ly/x9Usgb**), but basically we're going to need to be more considerate when building experiences in order to reach more platforms while still optimising for these brand-new hi-res screens."

Edward Cant
Web app developer
& designer
opticswerve.com

COMMENT

"The new iPad has the highest pixel density display available in a tablet. The increase in pixel density is managed in the browser via viewport scaling, with scalable designs faring well as a result.

"Be sure to take advantage of CSS rounded borders, gradients, shadows, etc, as these will make full use of the available pixels. Fonts are a scalable and efficient means of increasing the perceived quality of the design. Text that has been saved as an image will look pixelated in comparison.

"Hi-res image substitution is an option, but it has to be carefully balanced with the potential expense in terms of bandwidth, server requests and asset maintenance The device-pixel-ratio media query is worth investigating for further CSS tweaking. These are the core ideas. They are completely optional, but there are definitely optimisations available for those wishing to gain an edge."

Haptic technology – get touchy feely

A feature touted to make it on to the new iPad was haptic technology, but unfortunately this looks to be one for the future. Now we know it didn't make it, what is it? Effectively, it is technology that creates the illusion of texture via an ultra-low electrical current.

Senseg (**senseg.com**) is a technology company that 'has re-imagined and realised a new generation of haptic technology that delivers sophisticated tactile sensations and enriches digital communication'. They revealed that they had been in recent talks with tablet manufacturers, but could not be persuaded to divulge who.

The company's technology transmits electro-vibration stimulus via an ultra-thin durable coating on the touch interface that outputs tactile effects. The tech is not here yet but when it does arrive it will take the web to new heights. Backgrounds could actually have texture, navigation elements could be raised to create the illusion of 3D – maybe a new solution for accessibility? Text could emulate brail for blind users, games could gain a new dimension giving greater control, and hidden elements could be integrated into a page that can only be found via touch. Check out this video for a demo of how haptic technology will work: **cnet.co/vZyeml**.
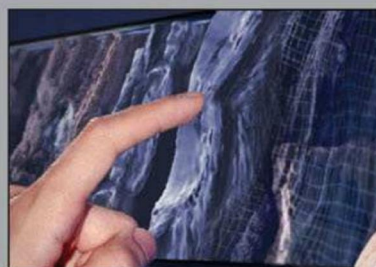
What is 4G?

4G is the successor to 3G, and is the next-gen tech behind superfast mobile broadband connectivity. Currently, there is no single standard for 4G; it is essentially a buzzword that incorporates a few different new technologies including LTE (Long Term Evolution) and WiMAX. Theoretically, 4G has speeds of 100Mbps, but in reality this is more likely to be around 12Mbps download and 5Mbps upload depending on coverage. In context this is around ten times faster than the current 3G standards. Though up and running in the US, 4G coverage is not available in the UK yet and is unlikely to make an appearance until mid-2013 at the earliest.

There are a few downsides – the most obvious being the lack of availability. Coverage is likely to initially be poor as new infrastructures will need to be put into place. The major cities will be the first port of call with outlying areas eventually catching up.
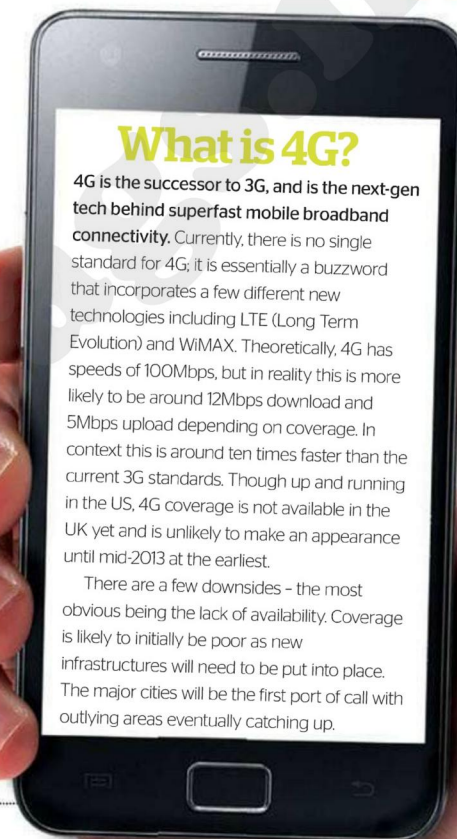
Crowd source

Your emails, tweets, forum comments. The social network but in print

This month we're discussing the power of Google Analytics, the merits of Dreamweaver and a social network to rival the heavyweights

Should Dreamweaver be your go-to software when starting on a career in web design?

Subject Loving web development
From Andy Wooldridge

Nowadays, being a web designer is more than just designing a few pages with HTML and CSS. There is the need to know at least the basics of web-development technologies such as jQuery and PHP, etc. So I must thank you for the addition of your new Web Developer section. However, can we have some top-quality PHP tutorials, as I would really like some professional guidance on learning this language.

web {developer;

The new section adds more breadth to the **Web Designer** brand and we are looking to cover as many relevant web-related development technologies as we possibly can. It goes without saying that included in that is of course PHP and, at some point in the very near future, a selection of PHP tutorials will be making its way into the new Web Developer section; watch *that* space.

Web Designer tutorials

Discover the latest techniques to help create inspirational and interactive webpages

Customise WordPress themes with widgets
bit.ly/AiMp7F

Learning JavaScript: Arrays and functions
bit.ly/xcuTNz
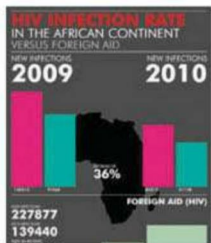
Animated infographics with HTML, CSS & jQuery
bit.ly/rP6Xl4

Subject Become a Google Analytics expert
From Wes Burnham

The Google Analytics feature (issue 194) made me realise exactly how important the tool is. It seems to have untold treasures hidden away that I didn't even know of. I know it's not the only metrics tool out there but it is the most popular and I would like to learn more about it and maybe even get myself some sort of accreditation. Do you have any recommendations to go about doing this?

Analysing data is a crucial tool in the process of helping a website get seen. The huge number of tools, especially in Google Analytics, makes it difficult to know where to get started. And then you have to wrap your head around what all those figures actually mean...

Google runs a host of courses and exams including the Google Analytics IQ exam and other related assessments such as the Google Advertising Fundamentals test. To get a taste of the IQ exam take a look here: **bit.ly/uB7L4n**. The tests typically last around 90 minutes and can be taken over a five-day period. To take the exam, a Google testing centre account is needed (**google.starttest.com**) and the only cost you'll incur is the exam itself at $50. Hope this helps.

Subject Is Dreamweaver a cornerstone of web design?
From Larry Mavis

I have just started to look into web design as a hobby, hopefully to become a full-time occupation. I have done some research on the internet and it looks to me that Dreamweaver is one of the tools that professionals seem to use. Is it the best software to get started with, and what other apps should I be looking at?

Dreamweaver is a program adopted by many pro web designers, Larry, and not without good reason. It offers all the tools needed to both design and build a website. However, many designers prefer to hand-code, which allows them more flexibility with how individual pages can be constructed.

It's not cheap and it will have a steep learning curve, but it can be a very useful companion when starting out. Also get Photoshop and you will have a powerful tool duo for the web-building process.

Subject Is Twitter still number one?
From Sarah Kilgannon

Social networking is dominated by Facebook and Twitter, and I use both... but I am looking for an alternative that is not populated by a host of idiots talking nonsense! I want to find a social platform that is directed at more creative people – any ideas?

There are many alternatives, but Twitter, and to some extent Facebook, are social platforms that offer plenty of pluses. When using Twitter be very careful who you follow and block anyone who doesn't fit in with your criteria.

The current 'in vogue' social media platform is Pinterest (**pinterest.com**). This is effectively a virtual pinboard, which lets users organise and share things they like on the web. Apparently, Pinterest the

Join the conversation as it happens on Twitter @webdesignermag
Comment on the news and opinion www.webdesignermag.co.uk
Email the editorial team at webdesigner@imagine-publishing.co.uk

platform is used to 'plan weddings, decorate homes and organise favourite recipes' among other things.

Subject New iPad envy
From Andy Leveson

I have never been able to afford an iPad, being a poor student, but I finally have enough cash to buy one. I know the new iPad is out but how much better is it than the iPad 2 and the original iPad? I only really want one to test my site designs on the tablet and see how great they look. Do I really need the new iPad or shall I get an iPad 2 or even an iPad and save myself a few quid?

This is a bit of a loaded question, but if you have the money and are happy to spend it get the new iPad. That said, the iPad 2 is still a very good tablet and has a better display than its predecessor, the original iPad. Alternatively, an iPad 1 will be dirt cheap but still do a good job for site testing.

APPS MAGAZINE

If you want to know about the coolest apps on the planet then **Apps Magazine** is the go-to publication. It provides trustworthy and respected reviews written by industry experts. The latest issue includes a guide to streamlining the clutter from your life with apps, 75 essential applications for both budding and pro photographers, a close-up view of the new iPad and its upcoming apps, plus much more. Head to www.knowyourapps.com to find out more about the latest and greatest apps around right now.

Comments from the Blog

Web Designer is always keen to hear from its readers, followers and visitors. Here we bring together a specially selected collection from the last month...

The 'Create interactive animations from a Twitter feed' tutorial triggered a flock of positive responses...

Liam said:
It would be really cool to have a 'tweeting' sound to go along with the app – something a user could turn on/off if they wanted to. Great idea though!

Blake Clerke said:
I can't keep up. I have an iPad 2 & I don't plan on upgrading. Unless it's something like this: **bit.ly/waiKRh**

Bryan Hickey
@lenkendall good suggestion on auctioning inactive @twitter usernames in @webdesignermag! Let's hope they take this on!

Michael Purnell
The newest @WebDesignerMag is great – has a lot of HTML5 features #webDesign

Embrace all feedback

Steve is the sales and marketing director of Fasthosts Internet Ltd (**www.fasthosts.co.uk**), one of the UK's biggest web service providers, which was set up in 1999. It was named Global Solutions Partner of 2011 by Microsoft Worldwide.

Steve Holford

Engaging with online customer comments is so important

The vast majority of businesses today now recognise that it is absolutely essential to be proactively publishing material about themselves online. A company website is now generally accepted as the ideal way to increase awareness, as well as enhance marketing and customer communication. Similarly, social-media sites can be a great way to connect to potential and existing clients alike. However, what about material published online by their customers – particularly negative material?

Today, increasing numbers of shoppers are voicing their dissatisfaction with firms on the internet. The snowballing of social networks and mobile internet has given new power to the customer to communicate and collaborate with one another on consumer issues. Many business owners find themselves trying to decide whether it is wiser to engage with such material or whether to simply ignore it. New research suggests that inaction is no longer a viable option. Fasthosts' Online Reviews Audit of 1,300 UK consumers found that it's now commonplace for Britons to express their gripes openly online, most commonly on social-networking sites such as Facebook and Twitter, as well as online forums. The survey concluded that one in three consumers now admit to having posted a negative online review of a company, product or service.

While willingness to share frustrations online is equal across both genders, the trend is particularly common with younger Brits. Young adults are far more likely to publish their grievances; in fact, some 52 per cent of under-24 year olds have turned to the web in this way compared to 39 per cent aged 25-34, and 38 per cent of adults aged between 45-54 years. One in four of those older (28 per cent) vented their concerns online. Geographical location also appears to be a driver, with the issue most common in the south-east (38 per cent) and far north-east of England (37 per cent).

Perhaps most significant is British consumers' willingness to forgive and forget in the right circumstances. Some 84 per cent of consumers would forgive a company for mistakes and be willing to use it again if the firm engaged with their negative online review and discussed the issues with them. There is clearly a good opportunity for businesses to use the internet as a cost-effective means to identify and win back unhappy customers. This is a strategy that both web professionals and their clients can benefit from.

Alarmingly, however, it appears that few British companies are seeing the benefits of getting to grips with negative online material. Data from a further study of 400 small UK businesses, also by Fasthosts, revealed that few UK companies choose to interact with customers who publish negative material about them. Despite the prevalence of online complaints, only 12 per cent of small firms have ever engaged with an online complaint. Web designers are well placed to advise firms that engaging proactively with negative online material should be written into their online strategy.

The majority of small businesses are missing the chance to demonstrate their willingness to learn and assure consumers of their dedication and trustworthiness. By engaging with customers, it shows a business is customer focused and enables other viewers of these websites to see a response to a criticism that otherwise would be unanswered. Social-media and customer forums can be ideal for picking up on such cases, and facilities for capturing reviews from a firm's homepage will be popular with the public.

Graham Jones, an internet psychologist, has some useful advice for SMEs that are still on the fence regarding the issue. He says: "People who want to complain about a company are getting a sense of freedom and power as a result of social networks and feel encouraged to make such complaints. As a result, this is a growing problem for companies and is something they must do as a matter of routine. Interestingly, several studies show that when people have their complaints positively responded to they are more supportive of the business than they were before the complaint was made. Psychological research shows this is linked to the fact that when the complaint is responded to well, people feel as though they are being cared for, which produces positive emotional responses."

It's understandable that many business owners' first instinct may be to shy away from their customers' grievances. Web professionals should surely discuss this with their clients and encourage them to tackle it head on. With support from their web advisors, small firms can win over more Brits. Research now shows that addressing negative online material will better enable their business to improve their customer experience, retention and online reputation. Every enterprise that takes the time to explore online customer reviews will see the internet working even harder for them.

WORK BETTER ONLINE WITH AWARD-WINNING TECHNOLOGY

Everything you need to work better online from Fasthosts

That's not just a web-hosting package, a domain name, a virtual or dedicated server, broadband, hosted email or online backup. It's **award-winning technology** you can depend on.

The awards tell you that you'll get the performance your business needs to work better online. They also tell you that businesses of all sizes succeed with our technology. You might not be an enterprise, but you can benefit from enterprise-level technology at Fasthosts.

Find out how your business can work better online call **free** on **0800 6520 444** or visit **www.fasthosts.co.uk**

You'll find some great offers there too.

fasthosts™

hosting | domains | dedicated & virtual servers | reseller hosting | online backup

<webkit>

Furnish your studio with the latest gear, gadgets and gizmos

iSuit
TBC
www.digel.de/isuit

1 German suit maker Digel has branched out into digital with the introduction of its iSuit. It is a well-tailored jacket and trouser combination with a built-in charger, inside pocket for a smartphone and headphone support.

Pro iOS Web Design And Development
£35.49/$44.99
www.apress.com

2 This book is all about designing and developing websites and apps for the iPhone and iPad. Learn how to implement HTML5, CSS3 and JavaScript to build for the iOS platform.

Pixlr-o-matic
Free
https://market.android.com

A free photo-editing application that transforms snaps with vintage and retro effects. The app allows you to apply colour overlays to create different moods, add lighting effects to brighten or darken shots, pick a frame or border, and share with friends on Facebook.

Pragmatic Guide To Sass
£18.99/$25
pragprog.com

3 This book shows you how to take CSS to the next level. It starts with the basics and works its way up to a host of advanced topics. There are clear and concise code examples with explanatory text and a peek at the Blueprint CSS framework.

Veho Bluetooth headphones
£129.95
veho-world.com

4 A designer set of lightweight and portable Bluetooth headphones oozing style and quality. On offer is a 20m range, a foldable design, a touch volume control and soft-on-the-ear leather earpads.

HTML5 Elements table poster
From $15.99
alaramills.com

5 The beautifully crafted HTML5 Elements poster is inspired by the periodic table of elements – remember your chemistry lessons? This gorgeous infographic is available in a host of sizes and every designer should own one.

Web services
Save costs by outsourcing your admin to these great online services

YouTube Video Editor
From Free
www.youtube.com/editor
Enhance your YouTube uploads by combining videos, trimming, adding text and inserting transitions.

LessAccounting
From £19/$30 a month
lessaccounting.com
An online bookkeeping service that lets you spend less time on your books and more time designing.

HTML & CSS – Design And Build Websites
£19.99/$29.99
www.wiley.com

6 A beautifully produced designer introduction to the delights of HTML and CSS. This 512-page full-colour tome uses information graphics and lifestyle photography to visually explain the core components of web design.

Digital Artist
£6
www.digitalartistdaily.com

7 Our sister mag is an indispensable guide to all styles and genres of digital art. It delivers high-quality tutorials, graphic showcases and techniques for the diverse digital creative community.

SketchTime
£0.69/$0.99
bit.ly/wbezKq

SketchTime is a simple, quick and reliable app for sketching and writing notes. It offers realistic pen strokes, two pen style tips, fast switching between pen/marker/eraser, smooth zooming and panning, unlimited undo and redo, and photo import options. A fun little app.

> ❝ Our sister mag Digital Artist is an indispensable guide to all styles and genres of digital art ❞

TP-Link wireless router
£40/$67
www.tp-link.com

8 A portable 3G wireless router that measures a minuscule 100 x 62 x 16mm. Share a 3G connection wherever 3G is available, compatible with over 120+ 3G USB modems.

Mobostick
$89.95
hsti.com/mobostick

9 The Mobostick is a universal USB connector for Android users who want wireless connectivity to their files or documents on a smartphone or tablet. Use the Mobostick app and wirelessly stream files from a phone to any device with a USB port.

Fiio E6
£21/$28.99
www.fiio.com.cn

10 The E6 is a small and extremely portable headphone amplifier. It has a reported battery life of around ten hours, three built-in EQ options and boasts a miniUSB connector.

Samsung Galaxy Nexus
£498
www.samsung.com

11 The sleek and curvy Galaxy Nexus comes with the latest Android OS (Ice Cream Sandwich), a 4.65-inch HD display, Bluetooth 3.0 technology, a front and rear-facing five-megapixel camera and HD video recording to boot.

LIGHT BOX
Gallery of stunning web design

10 Years in Type

www.10yearsintype.com

Development technologies XHTML, CSS, JavaScript carousel

Designer **Craig Thomas**

www.taylorthomas.co.uk

Showcasing the best in type design, Fontsmith's tenth birthday microsite revels in typographic excellence

nyone that has a love of typography will be well aware of the Fontsmith type catalogue that has been arming designers with its innovative typefaces for several years now. As the font foundry's tenth anniversary approached, Fontsmith wanted to mark the occasion by building a microsite to showcase its work, and also to launch a limited-edition boxed set of booklets that tell the fascinating story of its font designs.

The microsite was created by agency Taylor Thomas which shares Fontsmith's love of type. Partner Craig Thomas outlines the approach taken: "While we needed to show all the spreads from each booklet, we were keen to avoid Flash and in particular any 'page-turning' code or plug-ins. We felt keeping it simple would help let the booklets speak for themselves and so opted for a linear JavaScript carousel. This worked for the pack shots as well as the booklets. The site was made with desktops and iPads in mind."

As a complement to the limited-edition booklets that the microsite supports, the viewer is treated to a clean design that lets the type, as the star of the show, shine through each page. The understated background ensures that, when opened and viewed, each booklet is displayed in its full glory.

Thomas concludes: "The content needed to be accessible and share-friendly. As such, while the site operates on one single page with a lot of images, each booklet has its own URL as well as searchable text and thumbnails for sharing on Facebook. These are small but important details."

❝❝ We felt keeping it simple would help let the booklets speak for themselves and so opted for a linear JavaScript carousel ❞❞

<Above>
• Fontsmith's anniversary is an important milestone for the company. The limited-edition boxed booklets showcase the foundry's work

<Above>
• The font booklets are pieces of art in their own right, but Taylor Thomas's site design ensures they shine online as well as in print

Got it? Flaunt it! Tell us about your site ◎ webdesigner@imagine-publishing.co.uk

LIGHTBOX

Fontsmith

10

10 Years In Type is a celebration of a decade of work by Fontsmith, a window on the stories that have helped shape some of the world's leading typefaces and brands.

READ THE STORIES

www.fontsmith.com

abcABC
1234567890

<Above>
• Used throughout the microsite, FS Sinclair is a technical type inspired by the ZX Spectrum and designed by Phil Garnham and Jason Smith

#F7F8F8	#1AB5BD	#F2F2F2
#354861	#262626	

<Bottom left, clockwise>
• Oversubscribed since it launched, the microsite makes the font booklets highly attractive to all designers

• The website allows you to choose the font booklet that you'd like to view via a clean and simple user interface

• If you're unlucky not to get the print boxed set, you can still view each booklet online

• Fontsmith understands that its type designers are of paramount importance. As a result, Taylor Thomas's microsite offers brief bios of the foundry's star font makers

LIGHTBOX

Know a site that deserves to grace these pages? Tweet us now @webdesignermag

Grayden Poper

www.graydenpoper.com

Development technologies Photoshop, Dreamweaver, WordPress
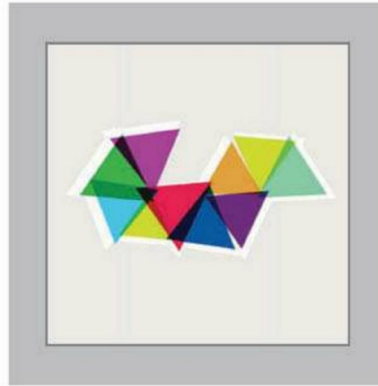
Designer **Grayden Poper**

www.graydenpoper.com

Interactive designer Grayden Poper's portfolio shows just how far WordPress's boundaries can be pushed
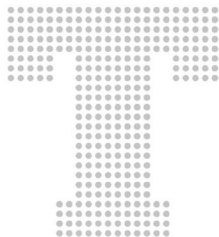
The strikingly simple design of Grayden Poper's online portfolio belies its sheer technical achievement. Using WordPress as the CMS, Poper's site positively shouts his personality. The last frame of the website says it all when he reveals his work ethic as: 'work hard, have fun and grow'.

Poper wanted to ensure his identity featured first and foremost. He reveals: "When starting this project there were two things I wanted it to accomplish. First was for it to be simple to browse, and second, it was important that my personality shone through the design, leaving a memorable impression on the viewer. I didn't want to create a site lacking personality."

This site could have easily become a chore with excessive scrolling, but Poper uses an innovative navigating system that enables the visitor to quickly jump to the portfolio item they are interested in. His work takes centre stage, of course, but with WordPress as its CMS, which is invisible to the user. As a result, the site is not only superb visually, but also a technical marvel.

What is clear from Poper's portfolio is that he has placed himself at the heart of each page. Not simply a showcase of his design, his site ensures that his identity is also in the spotlight. As he concludes: "The work itself might be amazing and memorable but the designer gets forgotten. So I did my best to create and provide a unique experience that would hopefully be memorable." We can safely say that this one stuck in our mind.

❝ It was important that my personality shone through the design, leaving a memorable impression on the viewer ❞

\<Above\>
• On the first page of Poper's site you're given an insight into what kind of designer he is

\<Above, top to bottom\>
• Poper says: "I felt it would be a good representation of who I am if I demonstrated my childhood's expectation of my future"

• As you scroll it's easy to forget this site is based on WordPress. Visitors can still leave comments by 'loving' the site via a simple click button

NAVY SEAL

I wanted to be a seal more than anything.

These are some of the things I wanted to be when I grew up. Instead, I ended up as an Interactive Designer. You can view my work and read my thoughts about design below. I hope you enjoy.

I AM OVER

abcABC 1234567890

<Above>
• Designed by Adrian Frutiger, the Linotype version is based on a group of fonts developed by Firmin Didot (1764-1836)

abcABC 1234567890

<Above>
• Originally commissioned by *GQ* magazine, Gotham can be obtained from Hoefler & Frere-Jones

#F7F7F7	#D0E51C	#31CDEF
#D2E61E	#6F237C	#E51C50

<Above>
• Between portfolio samples visitors get insights into Poper's personality. This elevates the portfolio into a much more engaging experience than many other examples out there

<Above>
• Poper's portfolio is comprehensive. It would have been easy to use a tile approach for navigation; instead users are treated to a fluid sidebar

<Above>
• Poper's approach to all his work is clearly communicated to visitors on the final page of his site

LIGHTBOX

Know a site that deserves to grace these pages? Tweet us now 🐦 @webdesignermag

2001- LAUNCH

inTacto Digital Partner is born

CONTROLES

Automatic   Manual

START

Mouse   Scroll
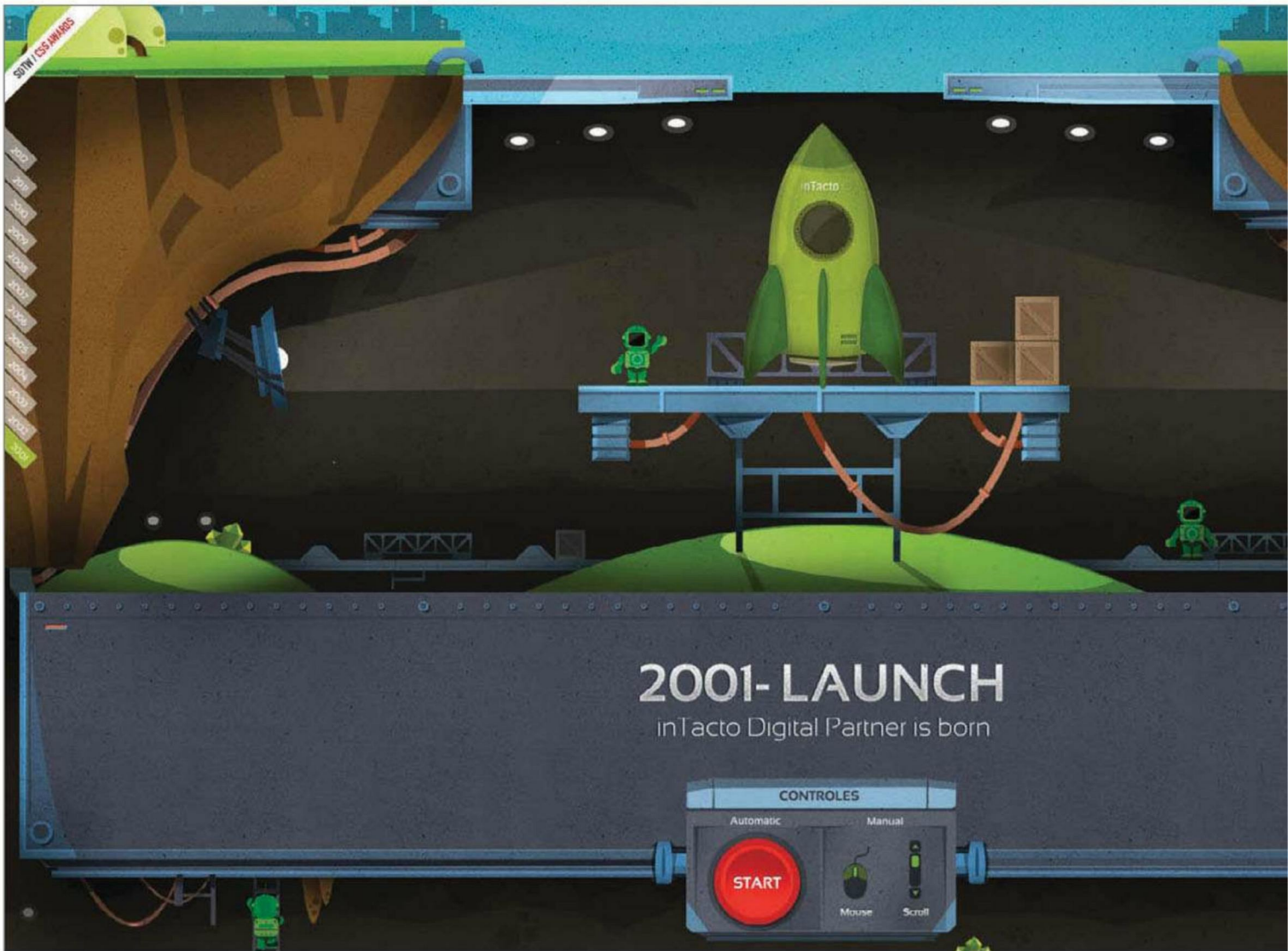
<Above>
• All systems go! You take control of the rocket ship that takes you through the history of the internet and how inTacto has evolved

HAPPY 2012!

2011- MISSION DECADE I
ACCOMPLISHED

BOARDING PLATFORMS

<Bottom left, clockwise>
• The eCard moves through each year and culminates with 2012 where inTacto ensures it connects with its clients and admirers through a host of social-media links

• Ah, 2001, when Netscape and IE6 reigned supreme. How things have changed since inTacto opened its doors

• Parallax scrolling is well implemented giving you the option of using the mouse or keyboard, or sitting back and watching the movie unfold in your browser

• Keeping up with the times is the core theme of this eCard. As each new technology came along, inTacto embraced it

#474A58	#97B401	#6790B1
#7B5429	#3E9EB4	#FF002A

inTacto 10 Years

www.intacto10years.com

Development technologies HTML5, jQuery, CC3, Photoshop, Illustrator

Designer **inTacto team**

www.intacto.com

This year's eCard was special, as inTacto wanted to celebrate its tenth anniversary by showing how both the agency and the web have evolved

Founded by Alejandro Lazos and Sebastián Caramés in 2001, inTacto is an interactive digital agency spanning both North and South America. Each year the company produces an eCard to showcase its work. But coming up to its tenth anniversary, something more ambitious was called for.

The inTacto team explain the motivation behind their new eCard: "We thought of different ideas and ways of illustrating the path we travelled and finally arrived at a good morphological mix between a space journey and stars and constellations representing internet explorers and milestones from each year."

As a calling card, inTacto's site offers an impressive array of technical expertise that is on show from start to finish. The retro styling of the graphics is timeless and illustrates that

this agency has a sense of humour. As the animation scrolls and the years pass, a clear evolution for inTacto is shown culminating in the ability of the viewer to show appreciation via their favourite social networks.

The use of parallax scrolling is an ideal way to guide the user through the story of the agency's development over the last ten years. According to inTacto: "We put in extra time and care in terms of details and quality in the design, and with good use of typography combined with the correct and justified employment of parallax, we managed to achieve a very interesting result."

Well-chosen fonts and superb graphics mix to create an effect that puts a smile on your face. Calling cards for agencies can often feel too 'business-like' when what is really wanted is an insight into the agency's sensibilities. This eCard shows what can be achieved when creativity and technical prowess combine.

abcABC 1234567

abcABC 1234567890

<Above, top to bottom>
• The main font on the eCard is Sansation designed by Bernd Montag; it is available from Font Squirrel

• The second-level copy on the eCard uses traditional Helvetica designed in 1957 and available in all weights from Linotype

❝❝ With good use of typography and the correct employment of parallax, we achieved a very interesting result ❞❞

Design diary

Ride the development cycle

Project | **Nike Chosen Series**
www.nikechosenseries.com
Company | **demodern**
Web | **www.demodern.de**

demodern

Background

Cologne-based collective demodern is granted creative licence to produce the highly acclaimed Chosen Series site for sportswear giant Nike

The world of extreme sport prides itself on a high-octane image that is characterised by a certain uncompromising attitude to life. The subcultures that exist around surfing, skating and snowboarding share a similar identity and outlook towards design. This in many ways is exemplified when an established sportswear manufacturer like Nike applies itself to these more niche pursuits and must communicate its message in a suitable way. For a project like its recent Chosen Series website, this involved harnessing the creative talents of an independent agency that understood the target market. "When Nike approached us and introduced us to the concept we were crazy about it and dove into it right away," explains demodern's UX designer, Jonas Wüllner. "The collaboration was superb; you meet so many likeminded people who love their job. Creating a concept for online communication almost becomes a breeze."

Chosen Series, for the uninitiated, is a snowboarding tour open to amateur riders only, which offers them the chance to showcase their skills alongside professionals. The official site would provide a hub for entry information, tour schedules and multimedia content from the series. This month we find out how demodern became involved in the project and also how much freedom they were granted to realise such a high-profile brief...

Rob Gurski,
aka 'The Design Weapon'
– designer

Jonas Wüllner,
aka 'The UX Mastermind'
– UX designer

Daniel Schroermeyer,
aka 'The Style Checker'
– art director

Recap winner concept drafts, produced with the good old-fashioned pen and paper

Daniel demonstrates the freestyle working approach adopted by the team
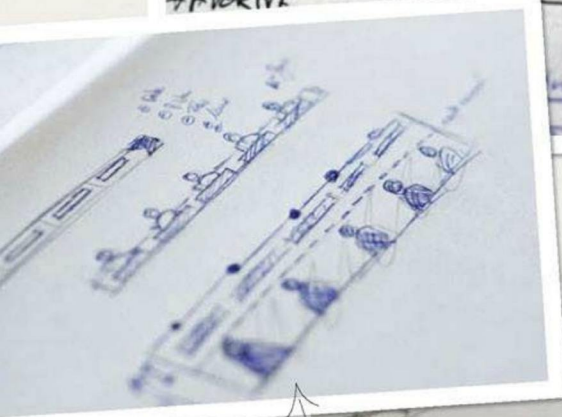
An example of some initial sketches for the site navigation

NIKE CHOSEN
SERIES
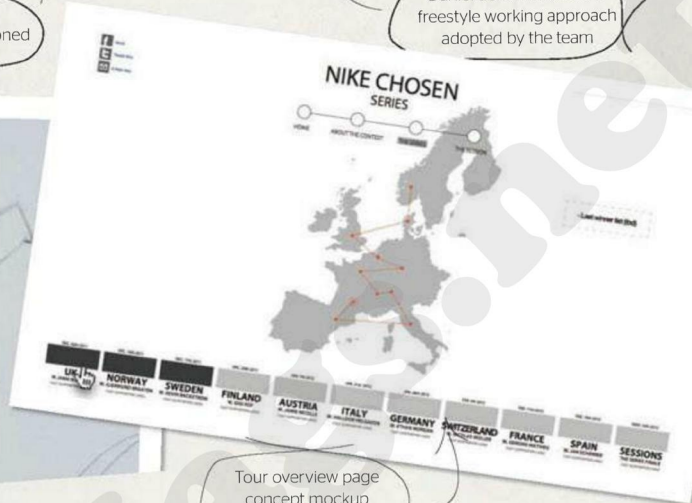
Tour overview page concept mockup produced by Jonas

DIARY STATS
7 PEOPLE INVOLVED

1 Concept

Visual brainstorming and a natural affinity for the subject matter helped in the beginning to form a vision of where the site should go

Jonas: "First I watched some snowboarding videos and did scribbles of snowboard jumps. Drawing allows me to think and I visualised the whole digital hub of the contest in my mind. Thinking about us telling the tale of a chosen one, it became clear pretty soon, that we would have to put the riders into the focus on this website. They are our stars – pros and ams at eye level. And the plan of the tour is at the same time the navigation of the page, so the single pages are arranged accordingly. Matching the tour navigation, the event pages also are positioned in a row next to one another, while the superordinate content stands for itself. There were no alternative interaction concepts in this project; we just knew right away that we

had found the right solution. Luckily Nike agreed."

Rob: "Basically there are still a lot of our initial considerations visible on the page. At the beginning there were relatively few assets already in place, which we replaced with sketches. At the same time we realised how well that new look fit the general appearance, so we abstained from using the 3D graphics."

Daniel: "You could say that we were optimistic while we did the first web layouts. The amount of imagery and text varied, and we realised we would have to aim for a large-scale but hugely flexible and modular layout." decision, because we later optimised the design for Facebook and mobile devices."

GO BIG OR GO HOME
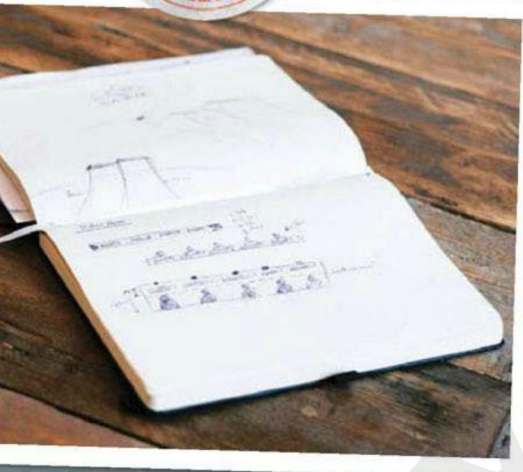
Early digital sketch for kicker, featuring typography eventually changed by the client

2 Design

Being given carte blanche to interpret the brief how they wanted was crucial when it came to designing the front-end graphics

Jonas: "**Sometimes it's easier to create something awesome when there are clear guidelines.** But Nike let us work at our own terms to the greatest possible extent. We decided against internal communication through Facebook alone via app/tab; our idea comes alive through staging the story. That is why we chose a microsite that we could design freely without being limited by the requirements of a third party (pixel measurements, technics, etc)."

Rob: "We create every page individually, so just doing a template and handing it over to the programmer to let him work out how to integrate the content is not how we do things. For that reason there is a Photoshop file for every single state of the page. Just designing the background patterns took a whole day, plus testing by the programmers – it is details like these that make the user feel the love for every detail, and which give depth to a page."

Daniel: "Most of the digital measures for Nike are being designed based on already existing campaigns with certain elements or defined style guides already in place that you have to abide by. In this case it was different; there were some visuals from the first campaign, but there were no concrete guidelines, so we were allowed to design relatively freely. Our talent for improvisation was requested, so we defined the style of the page ourselves.

"We also knew right away that we would closely collaborate with the development department, because we did not only focus on the look and feel but also on the usability and flow of the page. The whole user experience played a big role for us [in this project]."

Motion design typography animation screens showing the opening sequence being mapped out

Navigation sprite, showing photography and states for selectable riders and locations
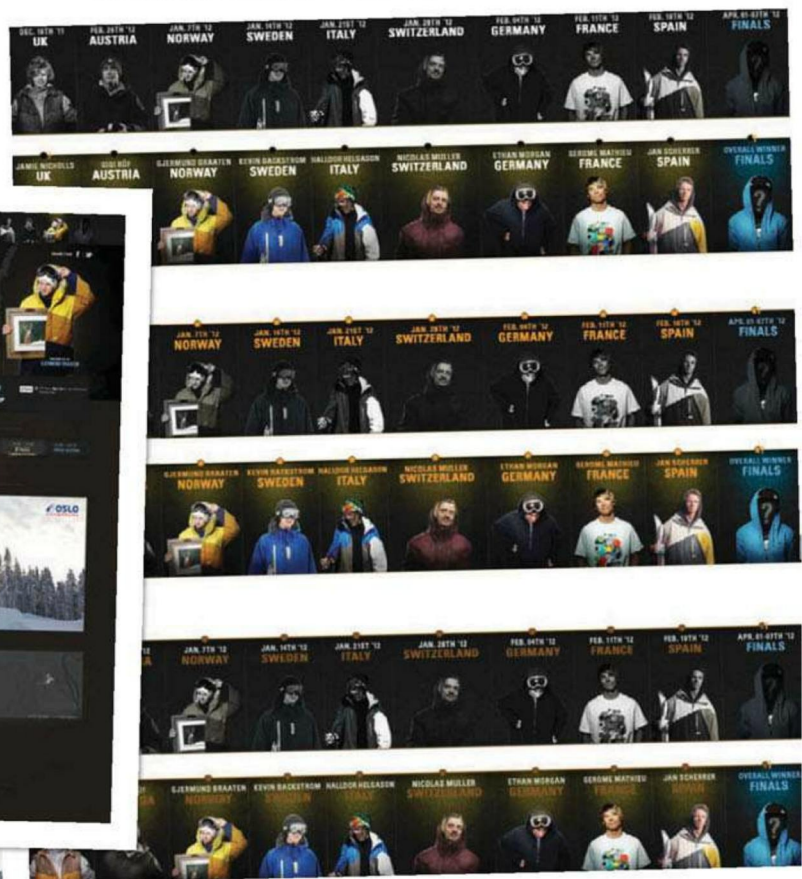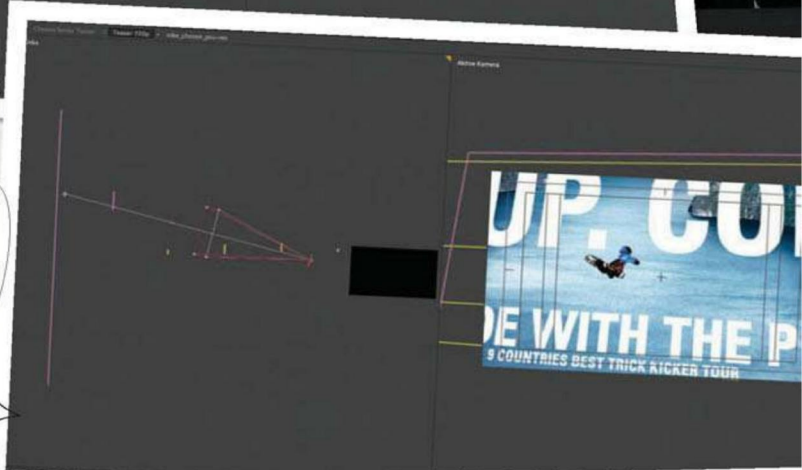
3 Build

"We still remain convinced that there is no need to blindly follow every fad"

One of the big decisions when it came to coding was the type of technology to use, with the team favouring a more next-gen approach, while tipping a respectful nod to Flash

Jonas: "We chose to use HTML5 and JavaScript, because, on the one hand, it works across platforms and, on the other hand, it is easy for us to [update]. Every week the page is being extended by fresh, additional content from national events, so we had to make sure that this would be easy and quick to do. The entire layout is basically one single page on which the content is spatially distributed, uploaded and moved around. Within the tour the movement between the pages even becomes visible. Through this the user has the feeling of actually moving the page from left to right. Nevertheless the entire script had to be very flexible. Dates could be moved at short notice because of weather conditions, for instance, so we had to be able to adapt the page immediately. Everything is based on our own demodern boilerplate including frameworks such as jQuery or Compass, which we keep optimising with every project."

Daniel: "Whether to use HTML or Flash was not a question that we pondered at any point because it has already become a standard to ensure cross-platform compatibility. But we still remain convinced that there is no need to blindly follow every fad, and therefore decided to use Flash for the intro, against all trends at the moment. Of course, this also comes with a fallback option for all mobile devices."
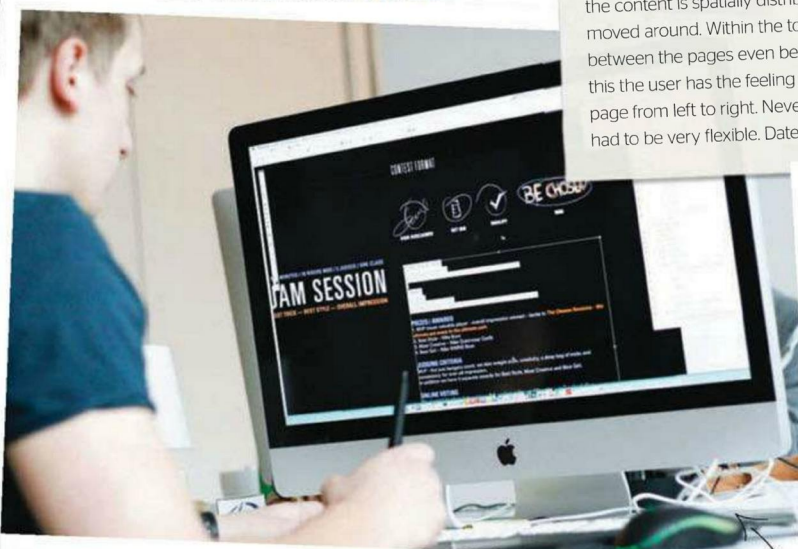
The launch

Jonas: "The biggest issue launching pages like this is the testing of Facebook functionality. Everybody at the agency has one or two fake accounts on Facebook to make sure that the testing environment does not get published every time the liking is being tried. But there is no such thing as a 100-per-cent guarantee, because you could, for example, realise that the text wraps differently after 100 or so Facebook likes. Therefore the lesson we have learned from this is not just to test everything for a day or two, but to also schedule an entire team for quality control after the launch. But ultimately those were only odds and ends. Since the page was honoured with a CSS award we are convinced that the bugs have been eliminated as far as possible. If not, please do contact us!"

Rob: "The timing was seriously tight; we had to prove that we were able to pull off everything we envisioned in a very compact timeframe, which we did. I was especially happy about the positive response from our partner agency, Urb-Orbis in London. If you guys are reading this: Greetings, we loved working with you too!"

Daniel: "There is nothing I have to add to that. Apart from maybe that the user's acceptance of the page is amazing as well. An average length of stay of six minutes is a lot. I am already looking forward to the finals, starting at the beginning of April. The next webpage for it is already online, just visit nikechosensessions.com.

Rob hard at work on info typography and graphics

The final launched splash page for the Nike Chosen Series website

Jonas logs into his fake Facebook account to carry out some sly tests

Promotional Facebook campaign page, launched pre-event

430 HOURS WORKLOAD
DIARY STATS

Lost and
FOUND.

Web Designer discovers the path taken by leading performance marketing agency Found. Core members of the team tell us about their experiments with responsive design, how they created the agency's online presence and the secret to their success...

who Found **what** Award-winning online performance marketing agency **where** Big Studios, 1 East Poultry Avenue, London, EC1A 9PT **web** www.found.co.uk

Key clients
Warner Leisure Hotels,
Dartington Crystal,
Reckitt Benckiser,
Goldsmiths,
Maplin Electronics

ound is a people-oriented PPC, SEO, social and mobile marketing agency based in Smithfield, London with offices also in South Africa. Launching back in 2005 as paid search performance specialist, Artemis8, the company chose to focus on generating incremental sales through the long-tail management of PPC campaigns across the retail, travel and finance sectors. Its unique selling point was built around a risk and reward model – to only be paid on results.
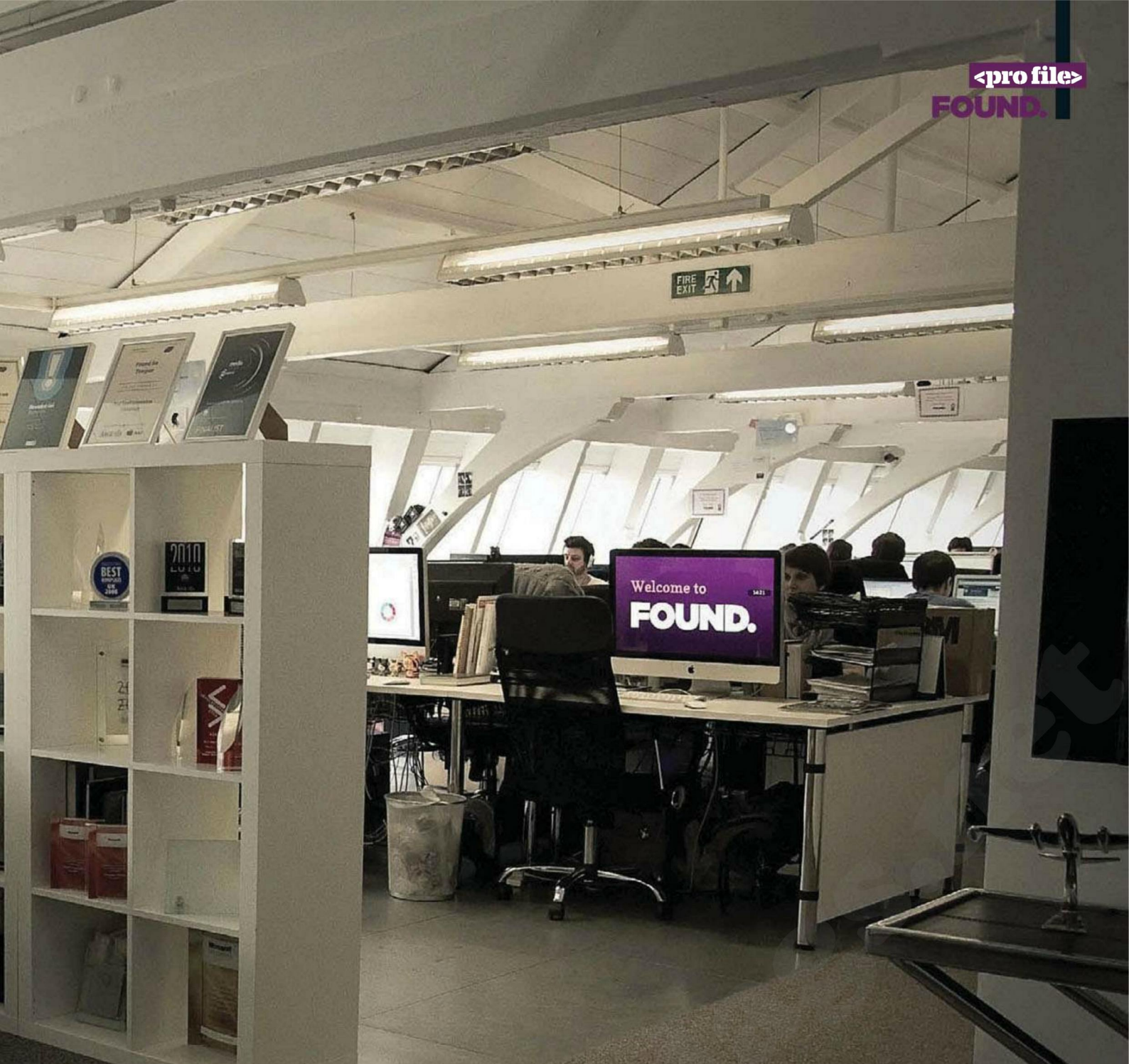
Brimming with a passion for excellence, the agency believes every client has a right to get the most out of their marketing investments, coupled with the highest quality of service. That's why Found has championed performance marketing since its inception.

Found understands that getting outsourcing right with any agency is essentially about accountability – something that can often get lost in retainers, management fees and percentage of spend contracts. That's why the business builds its services around its clients' demands and budget, as opposed to vice versa, taking the mystery out of what they are getting for their money.

Part of the Big Ideas Group, an enterprise that offers web design and build, website hosting, IT support services and represents one of the largest digital hubs in Birmingham, Found has delivered over £250m of sales value for its clients.

Found as a company has been around since 2010, but its foundations lay further back in the opening decade of the new millennium. Product director, Luke Townsend, explains the evolution of Found and how they got to where they are today. "Although we only officially rebranded as performance marketing specialist Found in November 2010, we have actually been a key player in the digital marketing arena since 2005, starting

life as specialist paid search marketing company, Artemis8. With the unique offering of only being paid on the results we delivered, we took the logical step to expand our offering to become a strategic performance marketing agency.

"Through our progressive rebrand to Found in 2010, we carried the performance legacy through to all the SEM channels we work in: PPC, SEO, mobile and social media, which means we now offer a cost-effective and scalable SEM solution, tailoring our charging model around meeting our clients' individual marketing objectives."

The brand name Found perfectly portrays the company's ethos. Townsend and lead designer,
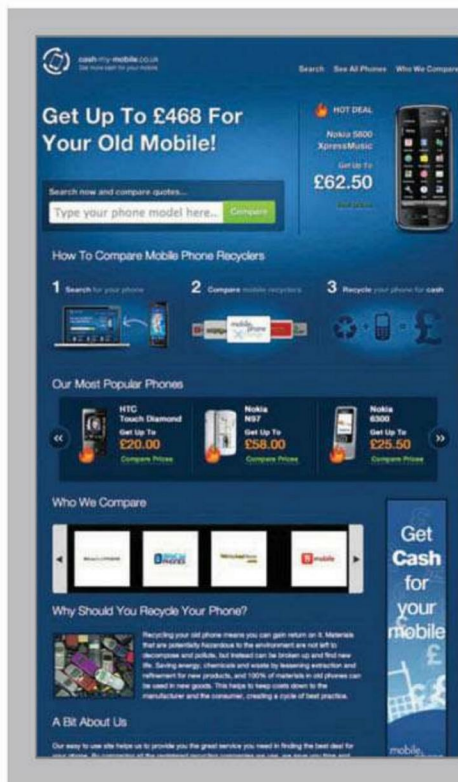
Five heads are better than one when it comes to plotting SEO strategies

Fun objects around the office make those stressful days a little more bearable

Found timeline

Established **2005**
Founders **Sebastian Gray, Lucan Gray**

Speaker at the NMA Live search event with a panellist slot alongside Google, Yahoo! and MSN.

Number of employees **18**

Performance Marketing Awards winner for Best Lead Generation Campaign for its work with Tempur.

Number of employees **12**

2010   2011

Launch party to celebrate the rebrand to Found.

Launched its SEO service offering, adding four dedicated SEO members to the Found team.

Tasked with managing the Goldsmiths and Mappin & Webb sizable SEO campaigns.

GOLDSMITHS SINCE 1778

❝ People are the heartbeat of our business, but client satisfaction is our lifeblood ❞

Andrea Downey, give an insight into the company name. "We wanted a name that stood out, a strong name. Found is a name that essentially matches what we do for our clients: we help them to 'get found' online, whether it be through search, through social media or through mobile [platforms]."

Townsend adds: "Having a name that is in line with our core offering is of huge benefit, not only in raising awareness of the agency but also in communicating what we do best. It was important to us that we had a name that we could build a strong cohesive brand around, and Found fits that bill perfectly."

The Found website is contemporary, clever and most importantly delivers the company's core message. Townsend and Downey reveal just how important the online presence is to the organisation.

Townsend says: "Company websites are often what give potential clients their first impression of you and determine whether they are interested in having a conversation and, in our experience, this is no different. We felt strongly that we needed a website that not only communicated what we do successfully and portrayed our company's personality clearly, but also showcased our talents as an agency.

"It has been built with SEO, PPC and social media in mind from the outset to enable us to harness as many visitors as possible while facilitating lead generation and conversions through our multiple contact points."

Downey tells us: "Our site is the face of Found – the first place a client is going to visit to find out more about us. We have worked hard to design and build a site that informs clients about exactly what we do, what we can offer them, which technologies we use and who we have previously worked with.

"WordPress was the perfect platform to create the site on, especially for blog postings and content updates. Our site is not just about the services we offer but also about the people who work here. Everyone in the team has their own page and has the opportunity to share their knowledge and opinions with the world. Usually there are three to four blog posts a week coming out of Found HQ."

Attracting clients is essential for continued success. Townsend reveals how good work, award wins and word-of-mouth get Found noticed. "Great work often leads to new clients. We've been very fortunate that our work with many of our clients has led to new

Lead Designer
Andrea Downey

industry insight

Luke Townsend,
product director,
Found

" Utilising technologies like geolocation and click-to-call functionality can be the difference between a successful campaign and a dud "

Crosby Street Hotel

maplin
The electronics specialist

Chosen by Maplin as exclusive PPC agency across desktop and mobile.

Winner of three awards for its search, mobile and affiliate marketing work at DADI and UK Search Awards.

Number of employees
35

2011

2012

Number of employees
27

Panellist at the a4uexpo speaking on the future of search marketing.

expo

Launched the new Found website and blog after many months of planning.

Speaker on mobile marketing at the UK Search Conference.

UK>SEARCH
>AWARDS>2011
>WINNER

---

business through expansion of the services we are providing to our existing customers. Word-of-mouth referrals and awards certainly help to increase our profile and attract new prospects too. That aside, we also practise what we preach and have created a content-rich website to help prospective clients understand what we do and how we can help them. We drive traffic to the website through paid search advertising, search engine optimisation and social engagement. Equally our blog has contributed significantly to visitors and enquiries, thanks to the frequency and variety of posts."

Winning awards is a great indicator of quality, but it's not the only sign. Downey explains how Found generally gauges the success of a project.

"Awards are fantastic accolades to have – from a seal of approval to a pat on the back to the team who have worked their socks off to deliver innovative ideas and impressive results. To win an award represents a

nod to all of the company, our clients and the wider industry, and simply spurs us on to more successes.

"Success can be measured in many ways though: helping a client exceed their targets; client renewals; client service expansion; and new business wins as a result of referrals. People may be the heartbeat of our business, but client satisfaction is our lifeblood."

The Found toolset is packed with open-source, bespoke and Adobe tools. Townsend and Downey reveal the software that makes the company tick.

Townsend says: "We use a vast array of tools to deliver strategically productive digital marketing campaigns. This may be utilising the technology available through the web or our own bespoke toolset. Daily, we use tools for keyword generation, ad position monitoring, link checking, click tracking, traffic analysing, A/B testing, phone call tracking and much more. We also use social marketing and search intelligence tools to give us an edge over competitors.

Downey continues: "From a design perspective, we use the same tools as any other agency: Photoshop for web design, Illustrator for print, Dreamweaver for small HTML/CSS site builds. We are also big fans of Basecamp as our internal project communication tool and version control software for all website work."

A wealth of search executives, search analysts, web designers and developers are just some of the key players in a project. The Found colleagues explain how the project workflow typically plays out.

Townsend begins: "Projects usually start with the search analysts doing what they do best: researching and analysing the search space, identifying the client's current weaknesses and where they're leaking traffic to competitors. Once our strategy has been identified and approved then we enter the studio, where we would look to start some early design concepts.

"This may take the form of creating a conversion-focused landing page, reworking a contact form,
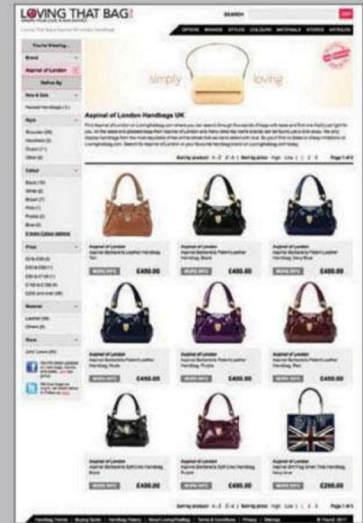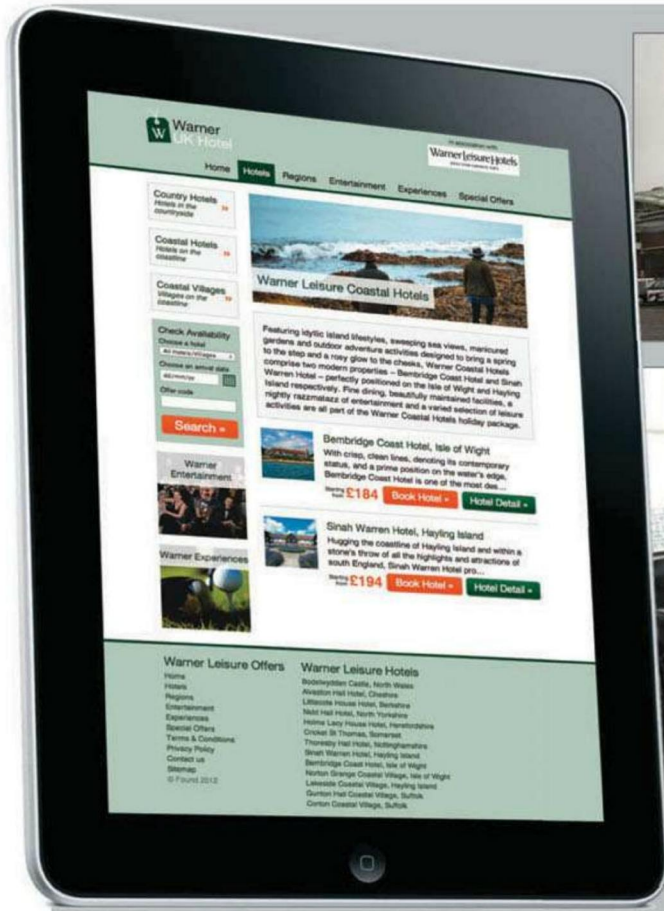
The historic Poultry Avenue offices where Found works its magic

Listening to, monitoring and engaging with social media is becoming more and more fundamental to generating traffic

creating a Facebook page or building a microsite designed specifically for PPC traffic."

Downey goes on: "Once the target pages are deemed ready for traffic, it's back over to the search team to build out the keyword lists and generate the ad copy and the process of optimisation begins.

"After launch, the keywords that drive the most revenue are identified, stripped out and given special attention, maximising their return by monitoring their positions in the search space closely, tailoring individual ad variations and designing bespoke landing pages. Studio plays an ongoing, if sometimes hidden, role in the success of many campaigns."

Projects at Found go through an evolutionary process that involves the clients. Townsend tells us how the projects work after the initial completion. "There is rarely a point at which the project is handed over and we dust our hands of it. Our search campaigns evolve, grow and require constant fine-tuning to get the most out of them. The same can be said for the supporting design work; a little tweak here or a change of layout there can have a significant impact on the effectiveness of a campaign.

"The Found team works very closely with our clients and any project is delivered as a partnership. Clients are, therefore, involved in the evolution and development of the campaigns and sites we are nurturing for them and the end-product is never something they have not seen or been involved in helping to create somewhere along the line."

SEO and PPC are two key services that Found offers to customers and clients. Townsend explains the core differences. "PPC and SEO are very different; paid search represents instant paid-for advertising. PPC provides immediate access to a huge audience at controlled costs. Through sound execution and solid optimisation, PPC not only reaches the audiences seeking you out but also opens up companies to new audiences who may have never even heard of them.

"SEO, on the other hand, is space that can't be instantly bought. With Google's ever-changing algorithms, SEO is a much longer-term process, requiring discipline and a solid strategy to achieve positive movements in page rankings. Having a dual strategy in place is certainly very powerful."

Social media is an integral component of any digital campaign. Townsend and Downey tell us just how important their role is in the generation of traffic.

Downey explains: "Social media is a major part of any website – who could imagine life without Facebook or Twitter? Users want to interact with companies, share their experiences with people, retweet blog posts and engage in online conversation. Search engines are picking up on this and it now contributes towards search ranking – great for SEO."

Townsend elaborates: "Listening to, monitoring and engaging with social media is becoming more and more fundamental to generating traffic. With search engines increasingly weighting their results based on links from social media sources, it's also permeating into the other areas of SEM – ignore it at your peril!"

Have you seen this man? It's been so long we wouldn't advise sending out a Google search party...

...min Briggs

...the merchant ship Mary Celeste ...the Atlantic Ocean near the Strait of Gibraltar ...ing since 4th December, 1872

FOUND.

Luke Townsend,
Product Director
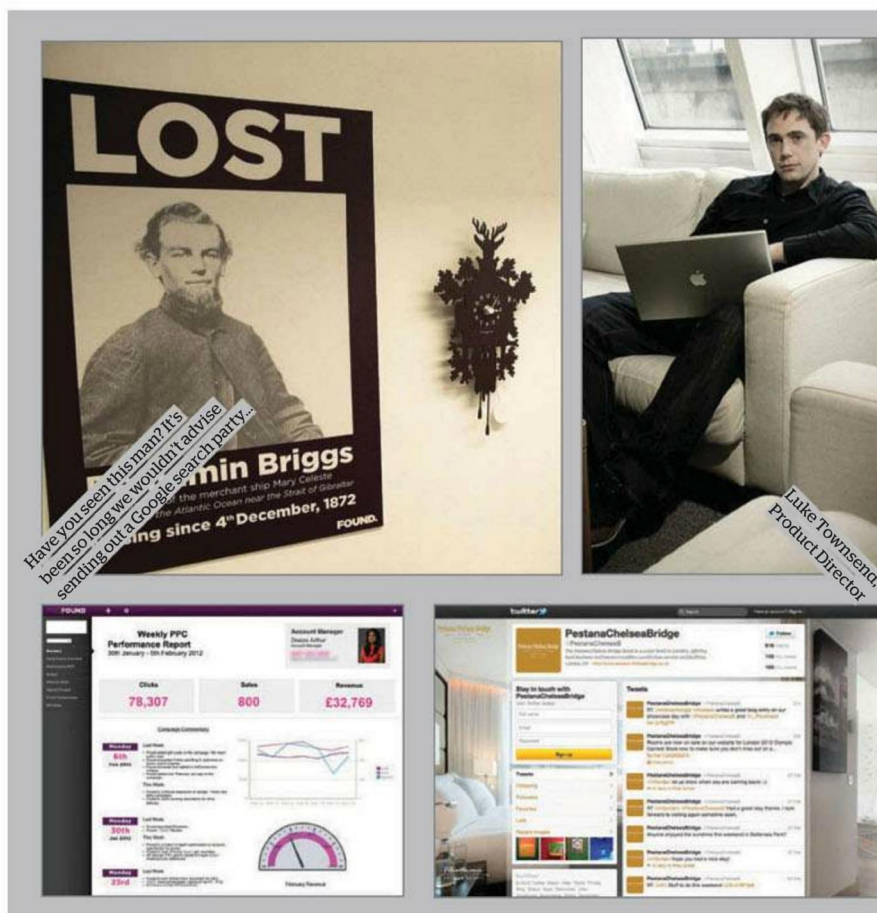
KEY PROJECT

Autoglass

How Found went about cracking the mobile site of this windscreen repairer

Found built and launched an extensive mobile paid search campaign, as well as a mobile-optimised site for windscreen specialist Autoglass within just four weeks. Together, Autoglass and Found pioneered click-to-call in the glass repair market and demonstrated a forward-thinking response to the rise of smartphone usage, mobile search and the rise of consumer demand for a simple way to contact a repairer in an emergency from the roadside. This project achieved several aims, including:

• Mobile PPC ads targeted by device type
• In-depth geo-targeting to deliver relevant contact details for local repair centres
• Dynamic phone number insertion tailored to referring search term
• Over 20 per cent average conversion rate.

Smartphones and tablets are central devices in today's web experience. Townsend and Downey explain how Found caters to the small screen.

Townsend begins: "While the principles are the same, the application needs to be tailored specifically for each platform. Utilising technologies like geolocation can make the difference between a successful campaign and a dud. Some campaigns naturally lend themselves better to mobiles and tablets, while others are much harder to get traction on these platforms. The key is to [approach] each campaign accordingly."

Downey continues: "Advertisers are waking up to the theory that although smartphones and tablets provide the same remote access to online, they are separate platforms and need to be treated as such. We're therefore seeing more and more sites having a desktop version and a mobile/iPad version, and cleverly utilising the strengths of each platform. However, the industry is now buzzing about responsive design, so we have started experimenting with this and have a few projects ready for launch that adapt on all devices."

Lastly, we ask the Found employees their thoughts on what it takes to get ahead in such a

notoriously competitive industry. Townsend replies: "Succeeding in online marketing is a science. We invest heavily in employee learning opportunities from training courses to regular 'lunch and learn' sessions.

"Found is a people-oriented business and, first and foremost, we look for passion, hard work and all-round ability. We want people who are skilled and always striving for the very best in everything they do. After all, we're a business remunerated on results so everyone has a role to play, and working as a cohesive team makes the delivery much more enjoyable and rewarding."

In conclusion, Downey adds: "We work in a great industry which has the advantage of having numerous free tools on hand to experiment with – from design and development to analytics and optimisation. There are so many blogs out there with people willing to share advice that it's possible to build up some experience before applying for roles. Internships are a great way to enter the industry; a number of our interns have become fully fledged 'Founders' following their internship. I would advise people to take risks and be bold; you've got nothing to lose."

FOUND.

WEB	www.found.co.uk
FOUNDERS	Sebastian Gray
	Lucan Gray
YEAR FOUNDED	2005
CURRENT EMPLOYEES	35
LOCATIONS	London, UK
	Cape Town, South Africa

SERVICES

> SEO
> PPC
> Social marketing
> Mobile marketing
> Search re-targeting

FOUND.
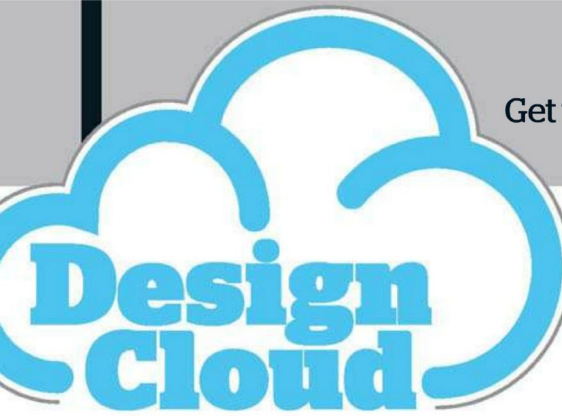
Design Cloud

Big beautiful backgrounds

Full-screen background graphics add zest to a web presence. We look at 24 backdrops that enhance the page and engage the viewer

RESPONS
DESIG

warface
★ ★ DIGITAL DESIGNER ★ ★

SIVE GN

As the devices we use to access the web continue to diversify, with all manner of capabilities and screens to support, the need for adaptive websites has never been so pressing, insists Kayla Knight

O nly a couple of years ago, the concept of responsive web design exploded on to the scene in what had been considered at the time 'the future of the web'. In the short amount of time since, so much has changed, and so much new technology has emerged. Now, it's quite obvious that responsive design is no longer just the future of the web; it is the practical solution that developers and designers should use today.

The notion of responsive web design is a simple one: creating websites that seamlessly respond to the user's environment, based on the device, size, resolution, technical capabilities, etc. If only the implementation were as easy as its ideal definition! Since Ethan Marcotte first introduced the concept into the mainstream web-design community in his May 2010 article in *A List Apart* (**www. alistapart.com/articles/responsive-web-design**), many articles, trends and new considerations have arisen. Debates are being held left, right and centre, and there seems to be a new best practice to adhere to or trend to follow for responsive design every week.

In this feature, we hope to fully discuss all of these modern considerations, see both sides of the debates and cover the most up-to-date methods for crafting responsive designs that work not only among today's devices, but those that can be made future-proof as well.

Ian Hart
The lead developer at Stink Digital looks forward to a time when responsive web design is seen as industry standard

Adaptable by default

I n the past year we've seen responsive design shift from an experimental technique used by individuals to a recognised collaborative process used by teams of designers and developers. This device-agnostic approach means that, rather than spending extra time designing a mobile-specific site, we actually build for mobile, tablet, desktop and everything in between.

For responsive design to really work, a designer and developer must work together closely. As the designer creates initial concepts/layouts and the developer integrates them, they react to each other's results and adapt the design or build accordingly. Used properly, this iterative approach produces a truly organic solution and helps designers and developers understand each other's processes.

The result for the user is an experience that doesn't deviate too far from one they're familiar with and perhaps caters to their context. They're not seeing a watered-down version of your product or service, rather the design is simply adapting to the capabilities of their device.

While 2011 was the year of sending a link to your colleagues and telling them to shrink their browser window, 2012 will be the year where responsive design becomes the norm. Soon, we'll just call it 'web design' again.

Designing for a myriad of devices

The basic concept behind designing 'for content out' is ultimately to move away from creating multiple solutions for every device instance. Certain rules and guidelines can instead by applied to dictate adaptable column layout, navigation, text, images and elements

Our technology grew too fast for us, and there are far too many inaccessible websites

THE IMPORTANCE OF RESPONSIVE WEB DESIGN TODAY

Several years ago, the 800 x 600 screen size was the most common, and the standard to design for. Shortly after, the 1,200 x 800 screen size seemed even larger, but is now one of the most common sizes. What the most common desktop or laptop screen size is today is only half the battle though, and rather irrelevant considering how often it changes. Plus, now we have tablets, netbooks, smartphones, large TVs and more that can all access the web.

The vast variation in screen sizes is likely what led the revolution of responsive web design to begin with, especially within the last few years. Of course though, along with all these new (and old) devices that are in use today, we have other considerations such as the technical capabilities of what these machines can handle.

It seems as though we can already be considered 'behind' on adapting our design and coding to the various devices out there; our technology grew too fast for us to catch up, and there are far too many websites that are inaccessible to a large number of users.

Many websites, especially those of larger companies who had the resources, tried early on to 'stay ahead of the game,' by creating perhaps several mobile versions of their websites, and even used the early concepts of flexible layouts to meet a wider variety of user needs. However, with the range of device considerations growing, this method of designing for each popular gadget to hit the market is becoming untenable.

Responsive web design creates a new possibility where we can cover, for the most part, all of these various devices now, and hopefully those of the future too. As more technologies emerge, of course there will be new considerations and practices for responsive design to grow upon.

Surely we don't have all of the answers now, but we can catch up our designs to meet modern needs, and improve the global user experience today. With responsive web design, we can also help future-proof our web experiences to an extent, and adapt far more easily when the new technologies do arrive.

BENEFITS FOR USERS AND DESIGNERS ALIKE

As if we needed more incentive for us all individually to get rolling on the responsive design trend, perhaps the biggest motivator can be that it not only creates a better web for our users, but also ushers in a better web for us designers and developers as well.

We can, of course, see the positive benefits for the users almost immediately:

• Users with devices new or old can have the same great experience on a website, eliminating some of the unnecessary frustration

• Users on the go or that would otherwise transfer their browsing to various devices can do so easily (eg a user that must switch from their desktop to their iPad while travelling)

• Locations in the world where technology is just catching up, and mobile devices are far more prevalent than any other for accessing the web, can take advantage of more information

• Businesses benefit by reaching a wider audience, and a well-built website for all users can increase a brand's trustworthiness. Also, businesses can save time and money in the long run on the maintenance of their website.

With all of these obvious advantages to the consumer in focus, we can often overlook the benefits responsive design can offer designers and developers too:

• More flexible designs mean less frustration, errors and bugs for different user situations. A design that adapts and, to some degree, 'fixes itself' from the outset means we can focus on the all-important user experience, rather than spending all our time correcting technical mishaps

• Complete design projects up front, rather than create new solutions for every new device size and capability. Relatively future-proof projects – at least, the near future

• The ability to improve our own design and development enterprises by being able to better aid users' and businesses' websites.

The above is only the beginning when it comes to the revolutionary advantages of responsive design. What's better is that all of these benefits are not a thing of the future, but so many practices can begin being implemented today by simply adopting a new way of thinking.

DESIGNING FOR 'CONTENT OUT'

Design is as important on the web as it is in any other medium, as a proper design can increase brand loyalty, conversions and add value. Despite design's benefits however, it's been a long-standing fact that content is still king and design only comes second. A good design will

Mark Boulton designs with his content, but it's surely not required to get an as effective content out design

flatter the content, and there is no reason why responsive design should be any different.

With the rise in responsive web design, we are beginning to see a new term: designing with the content out. By putting so much focus on content, we can create webpages that are easy to read

and use, no matter what their constraint (ie the device). Traditionally, we may use grid systems, common layouts and preset guidelines to construct a website's design. Later, we plug in the content. This isn't putting value on the content, but rather putting far too much value on the aesthetics of the website. With design out, content in, the content is not a priority.

With content out, on the other hand, we create the proper structure, width, size, placement and other typographical details that would most benefit the text, and then fit in the design based on the content's structure. Content out, design in.

DESIGN AND LAYOUT FOR THE WIDE SPECTRUM OF DEVICES

Designing for content out initially can perhaps be the simplest way to determine a proper layout for just about any range of devices. After we plan the content however, we still must think of where both the aesthetic design and other page elements will be placed. With responsive design, we are gifted endless possibilities. We can't possibly design separate interfaces for each instance, so we have to create a standard set of guidelines for user interface techniques to help simplify the creative process.

LAYOUT: COLUMNS AND WIDTH

Obviously, physical space will directly limit how many columns can fit into a webpage, but we shouldn't only worry about what's technically possible. Wider screens should use columns because it can be a great place for secondary content that will be noticed via our habit to horizontally scan pages. Even for smaller screens

edgeofmyseat.com's site uses horizontal space on wider screens to create a more useful spot for navigation and logo, but drops the sidebar to vertical stacking for thinner screens

Frameworks & boilerplates

It's great to know core techniques of responsive web design, both for the benefit of innovation and to build quality adaptive websites at speed. But there are plenty of frameworks and boilerplates available already to build responsive designs from. Below are just a few...

goldilocksapproach.com
The Goldilocks Approach is a responsive framework consisting of both CSS and JS files that tries to specifically step away from the responsive approach of designing for specific devices, and rather creates a fully flexible web design for each and every circumstance.

www.getskeleton.com
This is a minimalistic responsive boilerplate that has CSS and JS files to provide a flexible layout and highly compatible design elements, making it ripe for customising. There are several pre-styled elements included – all you need is your unique design flair.

bit.ly/saIAiW
Yamb is the responsive template from the Yet Another collection. The content-out boilerplate adheres to all the modern interface design methods for going from mobile sizes to larger screens. Features include a fluid grid, drop-down menus and responsive slideshows.

bit.ly/mRUHFs
Gridless is one of the most basic templates around. It comes with no predefined CSS grid classes or other designed elements. What it does include is an intelligent file structure, all the necessary CSS code and a JS file that increases compatibility in older browsers.

RESPONSIVE DESIGN
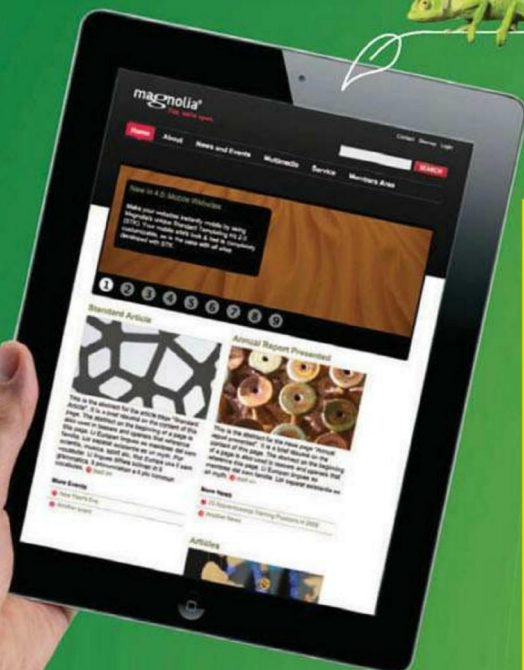
Boris Kraft
Magnolia's CTO explains how its new version 4.5 CMS instantly reformats your content for mobile devices

The responsive CMS

Having first introduced elements of responsive web design over three years ago, in our latest release **Magnolia CMS is refining this to new levels.**

All sites developed with our best practice Standard Templating Kit (STK) now benefit from an instant mobile version. However, for discerning corporate customers it is vital that the massive time and cost savings of this kind of automation do not come at the expense of maintaining the purity of the brand design and corporate image. This is why we have ensured that the instant mobile website can be refined for multiple different devices and fine-tuned with pixel-perfect control.

The other major lesson that we have learned is that it is not only graphical design, but also the relevance of content that is important. It is vital that the content itself is tailored and responsive to an ever-increasing range of client devices.

The high-resolution, big-screen video, detailed feature articles and granular menus that add real value to the user experience on a desktop can be irritating on a phone and potentially dangerous if I am trying to access the site from an IVI system. At the same time, companies don't want to have to build and maintain separate sites for an ever-growing spectrum of devices.

Content has to be developed and maintained from a central system but that system has to offer agile/flexible (depending on how buzzword compliant we want to be), multichannel output. This can be used to ensure that the type of content is responsive, not just to the interests of the visitor, but to the client device they are using, and even where and how they are using it.

The Magnolia CMS is a publishing platform that has aligned itself with responsive design features

on laptops, netbooks and tablets, additional side content should be put to use when possible.

On small mobile screens, beyond the physical space constraint, there comes a point where usability becomes a concern as well, and any side content should be rearranged.

FRIENDLY NAVIGATION

Navigation is arguably one of the most important elements of a webpage. With confusing or difficult navigation, the user may leave with a bad experience, or simply not even try. This is why it's essential to make sure that navigation provides the best experience for those on large screens just as much as it does those on smaller ones.

On larger screens, navigation that lies out horizontally is most common for a reason: it's a great space saver vertically, and is easy to scan. Vertical navigation is best for sub-navigation, whether in a sidebar or dropdown. As space becomes limited by width, we must alter the navigation so that it fits within a thinner area, but

is also user-friendly. So, horizontal navigation will often collapse into stacked button-like navigation when a browser size becomes thinner. There are several other things to consider when collapsing and expanding navigation.

Many smaller screens may be touchscreens, so use more white space (more touch-friendly buttons) – or switch to a drop-down select menu.

Hover effects are often irrelevant in touchscreens or limited mobile devices, so be sure to rearrange drop-down navigation accordingly or simply navigation where possible. Also, no need for decorative hover effects on smaller devices either.

For both large and small screens, consider usability for various input devices when designing navigation. For instance, users may be using a remote control (eg TV) to navigate, or the tiniest buttons (eg feature phone).

Navigation can surely be creative, and creative solutions are needed for responsive design. Navigation should be fairly simple in terms of

use, and effects limited for simpler devices. There's no need to put extra strain on old or incapable devices just for visual effects.

TYPOGRAPHY SIZE AND FORMAT

One big issue noticed early on in responsive design, and in just dealing with flexible designs in general, was the question of what the 'perfect' content width was. If a webpage with a large screen would have one piece of content span the entire width, it could surely become difficult to read quickly. On very small screens, we have the opposite issue, how few words per line can content have before legibility is sacrificed?

The ballpark figure for good readability on a screen is where text spans 10-15 words per line. Of course, depending on the actual content, language and so on, this number will vary, but the concept remains the same. For the modern computer screen size, this usually equates to about 600px in width, give or take. We cannot rely on exact numbers though, or on any particular formula for content width according to screen size. Instead, all we can do is what's practical as possible and use techniques for maintaining the 'content out' practice.

Below are a few general guidelines and best practices that are already being implemented throughout the web-design community in terms of content design:

• For larger screens increase font size, and for smaller screens decrease font size. Be sure there's a limit for both though, of course. Consider this for thinner columns too on large screens

Authentic Jobs collapses its mobile-inspired navigation to focus on the primary content: the jobs

51bits is a beautiful example of typography, and how type should alter itself in responsive web design

- Use legible typography, and allow extra white space for more detailed fonts, or for serif fonts. Follow basic typography best practices such as adhering to the baseline rhythm to help readability at any screen size
- Be creative with the use of drop-caps, intro or first-line text and so on. However, be sure that on smaller screens these added details do not interfere too much with readability within the limited space.

To make dealing with typography easier, there are a number of JavaScript and CSS techniques one can use to alter font details, size, and so on. However, be sure to consider compatibility with these extras if what's being implemented is necessary for the user experience.

IMAGES AND OBJECTS

Images have also received a lot of attention early on in the responsive design movement, and for good reason. They are a large part of our web experience, and just about any type of website relies on imagery. Likewise, the same applies to HTML objects – usually video.

We can make images and other block-level page elements responsive too. This doesn't only apply to making sure they are flexible in width, but also to be able to auto-crop or change proportions when necessary. Shorter images may be best for smaller screens, while our larger screens can take advantage of more vertical space for taller images. Images relevant to content, icons for navigation, or even logo versions may need to be replaced with different versions, depending on the space available. Or, perhaps any website dependent on block-level elements (think a portfolio website based on imagery) will need to change the layout entirely.

Excess imagery or video may want to drop out altogether for smaller devices to save space and loading time (there may be less processing power in smaller devices, especially older ones). For the video, images or other media that is kept on the page, it's a good idea to switch to lower-resolution versions based on screen size, or on detection of older devices as well.

The top graphic on illy issimo's site responds well to a smaller screen, maintaining its position while the main branding portion stays centred as it's resized

MOBILE APPS VS RESPONSIVE WEBSITES

When in doubt, or otherwise when in need of inspiration, take a look through some of the mobile applications available. These are built initially for the mobile experience, and if we can create our responsive websites to collapse down to be as user-friendly as these, then we've made some real progress. One downside of apps is that the user has to download them separately, and while at times apps are very practical, most of the time it'd be far easier to be able to have the same experience while mobile web browsing.

This is why we should always be thinking of ways to adopt the UI design techniques of mobile applications into our webpages at smaller resolutions. This may mean using more icons, drawing inspiration for navigation, or using similar design and coding techniques for buttons, forms, content display, scrolling and so on.

CSS3 MEDIA QUERIES FOR RESPONSIVE DESIGN

Figuring out the best layouts and design-related details may be the most challenging part of creating a responsive website, but when it's finally done, all that's left is making it all happen for the live site. Fortunately with some innovative thinking, most flexible and responsive design is possible with fully compatible HTML and CSS. The release of CSS3 media queries has had a huge impact on responsive design coding so far.

CSS3 media queries are what have made it to the forefront of what we as designers and developers can do to go beyond standard flexible designs with HTML and CSS, and add a whole new level of responsiveness to any website.

With CSS3 media queries, we can easily target specific devices such as tablets of smartphones, or better yet for responsive design, a range of widths and/or capabilities. Perhaps the most common use is declaring certain CSS rules for only a certain range in widths:

```
001  /* When the maximum viewing area is
     600px; this CSS will only apply to viewing
     areas thinner than 600px. */
002
003  @media screen and (max-width: 600px) {
004    body {
005      font-size: .85em;
006    }
007
008    .specificclass{
009      width: 100px;
010      float: left;
011    }
012  }
013
```

Responsinator

The Responsinator previewer simulates websites' responsive capabilities. It works as a series of embedded browser frames contained within illustrations of ten devices and adhering to the various dimensions and ratios. The usual suspects are supported, providing a reference on the various resolutions while you test your latest site's performance.

```
014  /* When the minimum viewing area is
1200px; this CSS will only apply to viewing
areas 1200px or wider. */
015
016  @media screen and (min-width: 1200px)
{
017    body {
018      font-size: 1.75em;
019    }
020
021    .specificclass{
022      width: 250px;
023      float: left;
024    }
025  }
026
027  /* It's possible to set a range too;
this CSS will only apply to viewing areas
between 800px and 1200px. */
028
029  @media screen and (min-width: 800px)
and (max-width: 1200px) {
030    body {
031      font-size: 1em;
032    }
033
034    .specificclass{
035      width: 200px;
036      float: left;
037    }
038  }
```

The syntax is pretty self-explanatory and easy to follow. Just begin with '@media', specify screen or print, and include whatever other queries the styles should abide by. Then, plug in whatever CSS is normally applied for that range in size. Any CSS rule applied outside a media query can be used for any screen size; properties such as backgrounds, global font styles and so on

shouldn't be re-declared in specific media queries, but rather globally for code re-use.

For a great beginner and more detailed guide on CSS3 media queries, check out **bit.ly/hHgltO**.

Below are a few more examples of queries that are useful for responsive design, with the same general syntax as above:

```
001  /* Layouts may change if screen is in
landscape or portrait proportions. */
(orientation:portrait) or
(orientation:landscape)
002
003  /* Specify by the actual physical
device screen width, rather than just the
current browser size. */
004  (max-device-width: 768px) or (min-
device-width: 240px)
```

Different stylesheets may also be referenced via CSS3 media queries, right within the link tag:

```
001  <link rel="stylesheet" media="screen
and (orientation:portrait)" href="portrait.
css">
```

While CSS3 media queries are definitely beneficial to use in responsive web design, we must always remember that some older browsers still don't support them! We can use JavaScript to add this functionality to an extent, and create flexible layouts via CSS2.1 as much as possible, but ultimately we must also consider graceful degradation for responsive design too. *Smashing Magazine* has a great and in-depth article on just that: **bit.ly/pOOqKZ**.

SCRIPTING FOR ADDED FEATURES

It's very important not to rely on JavaScript, or any other additional scripting to that extent, for the core functionality of a webpage, especially an adaptive one. Responsive design isn't only about adjusting to various screen sizes, but also about adjusting for the capability of whatever device – be it desktop, tablet or smartphone – will be used to view the page. There are many scripts that, if the technology is supported, can increase usability and work as an aid to responsive design. However, for those devices that have browsers old enough that they don't support JavaScript, or for device browsers that have it turned off, the webpage should still be 100 per cent user-friendly and adequately responsive.

However, there are a few ways we can use scripting on top of the core functionality of a responsive design, such as:

- Add further responsive/CSS3 compatibility for unsupported versions of browsers
- Help to alter content and media sizes, and to detect browser capabilities to add efficiency for less capable devices
- Add additional features that may be useful, just for fun, or provide visual bonuses for supported browsers on more capable devices.

There are plenty of scripts appearing all over the community that can add to our responsive designs and make implementing them a much easier aspect of our work.

Below are a few examples of scripts, complete with links that can be used to improve all of our responsive designs:

Response JS - A jQuery plug-in that allows for more flexible images, video and added efficiency based on device capabilities (**bit.ly/offOck**)

FitText - A JavaScript for 'inflating' web type. Adds flexibility to content by changing font size depending on screen size (**bit.ly/khGdEc**)

css-mediaqueries.js - Helps make some versions of browsers that don't support CSS3 natively work with it (**bit.ly/b1CO5P**)

Convert Menu to Dropdown - A jQuery plug-in that converts any list menu into a select dropdown with smaller browser widths (**bit.ly/z4CyDy**)

Masonry - A jQuery plug-in that can help layouts by stacking block-level elements accordingly dependent on screen size (**bit.ly/j3Z7Ie**)

FitVid.js - A simple jQuery plug-in for responsive video embeds (**bit.ly/nid1yz**)

Elastislide - A jQuery plug-in for a responsive carousel (**bit.ly/pYt151**)

Responsive-Images - A JS script that automatically creates responsive images by adjusting not only proportion but controls download size and adds efficiency for less capable browsers (**bit.ly/gO9bGo**)

CONCLUSION

Responsive web design is the future of the web, there is no doubt about that, but it is also the 'now' of the web. We as designers must learn these techniques and begin implementing them today, whether that means starting with our own personal websites or our very next project. As much as it's useful today to create a responsive website, it will be that much more necessary in just a few short years' time, if not before.

With all of the resources available already though, responsive design doesn't have to be time consuming, nor does it have to be frustrating. We have so many boilerplates, frameworks, scripts, snippets and tools at our fingertips that we can - and should - take complete advantage of. Beyond that, we can further innovate new trends and techniques ourselves with leading-edge creative thinking. This is just the beginning of responsive design; we can only imagine the new devices and technologies that will come about in years to come, and how this new strategy will evolve.

Learning to adapt

Sarah is a designer specialising in iPhone and iPad UIs. She works as both a consultant and a designer to various brands in the UK and abroad as well as speaking regularly at web-related events all around the world.

Sarah Parmenter

Responsive design represents a new way of thinking

Responsive web design has not just become a buzzword – it has heavily integrated itself into our best practice and workflow, but with it brings its own unique set of challenges. Our tools as designers, that have always been somewhat of a hack anyway when it comes to reforming them for web UIs, are not standing up to the fluidity needed to design for different canvas sizes.

It's something that has been on my mind recently; I've also blogged briefly about it. I'll be the first to admit, I was a bit behind on my responsive design skillset, as I'd had my head down in native iOS projects the last quarter of 2011. Since then, I've been thrown in head first with various projects and, not only is it a different way of thinking – switching to percentages instead of pixels and sometimes creating multiple sets of imagery – but it throws you into a tizz with your workflow as well. Static Photoshop comps are suddenly meaningless, and your process, budgets and timings all need to be reworked from what you knew as the 'norm'.

The reality is that many people need the creative process to happen in their chosen graphics editor – it's where ideas and happy accidents occur – rather than switching to thinking structurally about HTML. It's a very hard task to combine the two: creative thinking while essentially thinking in terms of bounding boxes. I've always found my designs come out that way – boxy – rather than with the layer of polish and creative flair I get in Photoshop.

Of course, everyone's creative process is different; there is no right or wrong answer, but it seems everyone I've spoken to is struggling with workflow. I've started to find it easier to establish what I call a 'visual style guide' at the start of the project, coding up buttons, list items, general headings and paragraphs on to one plain background HTML document and styling them using a hybrid method of Photoshop and CSS/CSS3 where applicable. This has worked quite well, as the client can see colours, typographical hierarchy and all the different elements that go into making up their page, without signing off the structural

> ❝ **Many people need the creative process to happen in their chosen graphics editor – it's where ideas and happy accidents occur** ❞

placement of anything. It also then serves as a handy document going forward where, if a developer is personally coding elements of other pages, they have a document for reference that should be as easy as copying and pasting in the relevant classes and IDs. This has worked nicely across a few projects, but I still find myself designing – at least the initial ideas for a general mobile, tablet and web layout – in Photoshop, which just isn't built for quick iterations to layout like that.

In response to my blog post, someone made a comment regarding designing the user experience, and how this is practically impossible in a static graphics editor. I agree, and that's when we can argue the semantics of 'what is design and what does that broad term encompass' until we are blue in the face. A layer of user experience is easier to design in the browser; it's much simpler to see how elements interact with one another when you can physically link them up on a page. However, I'd argue that this layer also gets designed offline first, as it's one of the first parts of a project for me; the browser just becomes a handy tool for accomplishing it – I'm not truly designing 'in the browser'.

There's no doubt about it, responsive (personally, I still prefer the term adaptive) layouts are the way forward. They break down the barriers between devices, in a world where so much content is now consumed on handhelds. As designers, we just need to find our feet with the best way to present these designs to clients and establish it firmly in our workflow. I think we're still in our infancy with this, and we need to think ahead to negate the 'it needs to look the same on every browser' syndrome that we developed years ago.

There is lots of progress to be made, not just with our tools, but also client deliverables, workflow and so forth. It's an exciting time, and I've heard on the grapevine that Adobe hears our prayers and might have something on the back burner to address them – only time will tell. In the meantime, I'll be sticking with my pen, paper, Photoshop, HTML/CSS technique, even if it is a slightly long-winded process. After all, if it ain't broke, don't fix it.

‹tutorials›

Responsive design with elastic text and images

Keep up with the times and adapt your web creations for mobile browsers

tools | tech | trends Dreamweaver or web editor of choice
expert Sam Hampton-Smith

On the CD
• Completed project files

Elastic Text & Images

The web is changing...

‹Above›
• Our basic HTML doesn't contain any special code other than some normal HTML5 elements

Web designers used to know exactly where and how their work would be displayed. Sure, we've always had the problem of supporting multiple browsers and all the workarounds that that involves, but in the past we knew that websites would be accessed from a desktop. Today that is no longer the case.

What does this mean for how we design our page layouts? Put simply, we can't rely on a visitor having a minimum screen resolution of 800 x 600px any more. We need to consider everything from an iPhone with a screen width of 320px, all the way up to a high-resolution display at 1,920px wide. This presents numerous challenges, not least to the way we display text and images.

Thankfully there's a straightforward solution to ensure our text is rendered at an appropriate, legible size no matter what the width of the user's screen. The same applies for images, and together these techniques form a useful part of an overall 'responsive web design' approach that enables our content to remain up to date and usable across all modern platforms.

In this tutorial we'll build a jQuery plug-in that lets us quickly and easily make text elastic, and then we'll examine some CSS to do the same for images.

> ❝ In the past we knew that websites would be accessed from a desktop browser – that is no longer the case ❞

01 The basic page

We're going to use a very basic HTML page to develop our plug-in. The code will work on a more sophisticated design, but it helps to develop without any unnecessary bells and whistles. First, open the 'start. html' page from the cover disc to access the full code.

```
001  <!DOCTYPE HTML>
002  <html>
003    <head>
004      <meta http-equiv="Content-Type"
content="text/html; charset=UTF-8">
005      <title>Responsive Design with elastic
text and images</title>
```

02 The initial CSS

We've already got our basic styles set up: a single-column design with different parts of our content sized to create a headline, intro/standfirst and then some paragraphs of body content. We'll leave this CSS pretty much as it is for the whole project. Grab the code from the CD to apply it.

```
001  /*
002    RESPONSIVE DESIGN WITH ELASTIC TEXT &
IMAGES
003  */
004
005  body {
006    margin:          0;
007    padding:         0;
008    background:      #333;
009    color:          #ccc;
010    font-family:     helvetica, arial,
sans-serif;
011    font-size:       62.5%;
012  }
```

03 Web fonts

One element our design is missing is a nice looking headline font for the top of the page. Let's deal with that really quickly by grabbing an appropriate typeface from Google Web Fonts (**www.google.com/ webfonts**) and applying it to the <h1> in the CSS. We have opted for Chewy and added a drop shadow to the text to add some texture.

```
001    <link href= http://fonts.googleapis.com/
css?family=Chewy' rel='stylesheet' type='text/
css'>
002    h1 {
003      text-transform:    uppercase;
004      color:            #fff;
005      font-weight:      normal;
006      margin:           0;
007      padding:          0;
008      margin-top:       30px;
009      font-size:        10em;
010      font-family:      'Chewy', cursive;
011      text-shadow:      10px 10px 0px #000;
```

<Above>
• This highlights the problem we're trying to overcome. When the page is viewed on a small screen the font size is all wrong, making the content essentially unusable

<Left>
• Loading in the CSS shows our design is a single column with a couple of headlines, an intro and some body text below

04 Approaching the problem

The key problem we're trying to address is ensuring that text remains appropriately sized when the window is resized to a larger or smaller width. The heading font, for example, might look great at 4em in a desktop browser, but massively too big on a smartphone screen. We'll use jQuery to help us set the font size according to the size of the window.

05 Prototyping

We'll start off by adding an event listener for the window being resized, and inside this listener we'll add some code that analyses the width of the window, then uses it to calculate the size of our text, using a proportion of the minimum and maximum that we'd like to display at. Grab the code from the disc and add it to a JavaScript document to get started.

```
001   $(document).ready(function(){
002       maxwidth = 1200;
003       minwidth = 480;
004       parafontsize = 1.4;
005       // Add your interactive code here
006       $(window).resize(function(){
007           width = parseInt($(this).width());
008           if (width<minwidth) width=minwidth;
```

06 Some basic maths

We've already set up variables to define the min and max width, and we've restricted our width variable to fall between those two extremes. Now we need to use this width to calculate how big our fonts should be.

```
001   pfontsize = parseFloat((width/
minwidth))+"em";
```

07 Assigning the change

We've got a calculation for the size of our text expressed as a percentage of the base width, so let's now apply this to the .intro text by using jQuery's CSS() method to target and set the font size property.

```
001       if (width>maxwidth) width=maxwidth;
002       pfontsize = parseFloat((width/
minwidth))+"em";
003       $("article p.intro".
css({fontSize:pfontsize});
004       });
```

08 Adjusting line height

When we preview our code it works nicely, but there's a small issue that becomes apparent as we go to extremes of text size. The leading doesn't suit the text at all sizes. This is because line-height is set absolutely in the CSS. To fix this, we need to set it programmatically.

```
001   pfontsize = parseFloat((width/
minwidth))+"em";
002   $("article p.intro").css({fontSize:pfontsiz
e,lineHeight:(parseFloat(pfontsize)+(parafontsi
ze-parseFloat(pfontsize)))+"em"});
003       });
004   });
```

09 Set a base size

Now we've got no real control over the starting size of the text. This makes the code less useful than it should be. We'll set up a new variable to act as a multiplier for the calculated result. So, starting with a value of 1.4, the final font size will be a multiple of 1.4.

```
001       parafontsize = 1.4;
002       // Add your interactive code here
003       $(window).resize(function(){
004           width = parseInt($(this).width());
005           if (width<minwidth) width=minwidth;
006           if (width>maxwidth) width=maxwidth;
007           pfontsize =
parseFloat(parafontsize*(width/minwidth))+"em";
```

Responsive design with elastic text and images
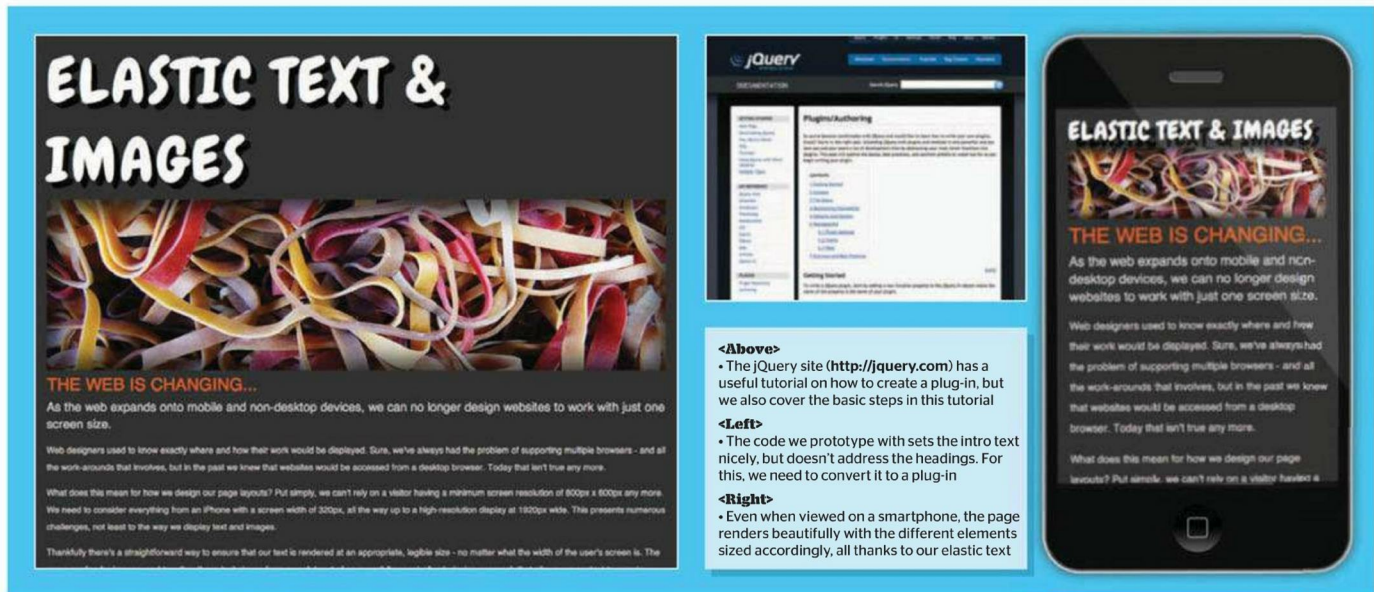
JavaScript turned off
Don't forget that users can turn off JavaScript. This means you should use commonly acceptable default font sizes for each element you set with JS code.

10 Review

We've now got a good chunk of code in place that will respond to the window size, fitting the font size and allowing our text to responsively update itself to the presentation view. It's all working as it should do, and across different browsers. There is a problem though...

11 Limitations

There's nothing wrong with the code, but what if we also want to change headings? We'd have to duplicate our code, which is messy and wasteful. Instead, what we need to do is turn our prototype code into a plug-in that we can efficiently call for any element.

12 Convert to a plug-in

Converting our code to a plug-in will enable us to call it on each individual element by simply chaining its name to the end of a selected element, abstracting the text-resizing code. For flexibility, we can set up our function to allow different values for the default multiplier, line-height and min/max width properties.

13 Structure of the plug-in

jQuery plug-ins look a lot like a normal string of jQuery code, but they have some extra bits at the beginning and end to create the function/plug-in, and to return the jQuery object. This is essential to support jQuery's 'chaining'. A plug-in looks like the code below.

```
001   (function($){
002       $.fn.pluginName = function(options) {
003           var defaults = {
004               property1: 1,
005             property2: "something
```

14 Assigning settings

Plug-ins tend to have a series of settings and options that determine how the function will affect the elements it's applied to. Our plug-in code accepts an options argument, so we need to assign these as a property of the function so they're accessible to our script. We can do this by adding this simple line of code:

```
001   var uoptions = $.extend(defaults, options);
```

15 Setting default options

However, we need to account for the possibility that when we call the plug-in on an element, we don't pass in all the options. We can generate some default options for the plug-in that will provide a basic result. We do this by creating an object for the default set of options before applying the settings.

```
001   var defaults = {
002       multiplier: 1.4,
003     lineheight: 1.4,
004     minWidth: 320,
005       maxWidth: 1200
006     };
007
008   var uoptions = $.extend(defaults,
options);
```

16 Convert the maths

We need to use the same formula we devised for the mockup, but apply it more generally using the plug-in settings. Copy and paste our original code and substitute the settings variables for the values we had previously. Your final code should look like the disc's.

```
001   obj = $(this);
002   width = parseInt(obj.width());
003   if (width<uoptions.minWidth)
width=uoptions.minWidth;
004   if (width>uoptions.maxWidth)
width=uoptions.maxWidth;
005   fontSize = parseFloat(uoptions.
multiplier*(width/uoptions.minWidth))+"em";
```

17 Output results

The final step here is to replicate our original code that actually sets the font size. Instead of using explicit calls to the elements in question, we'll continue to use the obj reference, and we need to substitute in the correct variable names for the font size too. Apply the code below to put this in place.

```
001   obj.css({fontSize:fontSize,lineHeight:(pa
rseFloat(fontSize)+(uoptions.lineheight-
parseFloat(fontSize)))+"em"});
```

18 Invoke the plug-in

Now we've completed our plug-in, we need to call it from our script. Bear in mind that we could call our plug-in code in from a separate JavaScript document, so although the code is there, we still need

to explicitly invoke it. Thankfully this is simple, as we can call it just as we do with any built-in function. Add the code below inside a document.ready() function.

```
001    $("article p.intro").elastictext({multipl
ier:1.15});
002    $("article h2").elastictext({multiplier:1
.8,lineheight:1});
003    $("header h1").
elastictext({multiplier:3.05});
```

19 ### Window resizing
If you test your page just now, you'll notice that it resizes the text when we first load the page, but as we manually resize the window nothing happens – the two do not grow together in size in sync. This is because our code only calls the plug-in once, and we need it to do this whenever a change is made to the window. We haven't yet done anything to handle the window being resized, so let's fix that now.

20 ### Overcoming the problem
We need to call our plug-in each time the window.resize() event fires. We can do this simply by adding a copy of our calls to the plug-in inside a window.resize event handler. Apply the code found on the disc to set this up, and then test the page again to ensure that it responds as intended.

```
001    $(window).resize(function(){
002      $("article p.intro").
elastictext({multiplier:1.15});
003      $("article h2").elastictext({multiplie
r:1.8,lineheight:1});
004      $("header h1").
elastictext({multiplier:3.05});
005    });
```

21 ### Elastic images
A quick way to get images to behave in an elastic way is to put them inside a container that scales simultaneously with the width of the browser window. If we then set the width of the image to be 100% using CSS, we can force the image to scale proportionally to the width, just the same as the text. Apply the CSS below to action this.

```
001    img.elastic {
002      max-width:        1200px;
003      min-width:        320px;
004      width:            100%;
005    }
```

22 ### Test and develop
Now we have a fully operational jQuery plug-in that allows us to create elastic text at will, you should test in all available browsers to make sure the multiplier values you've chosen work well. Consider how you could automate the discovery and setting of the multiplier values by querying the CSS for each element.

Code library
The completed plug-in
We've written our code as a jQuery plug-in to make it easy to apply to multiple elements. Here we break the function down:

To kick off, we create a set of default settings to apply to the plug-in if the call doesn't include any passed-in options.

The nuts and bolts of the resizing is done here. First we calculate the size to change the font to, and then we apply any changes to the styling.

When the page is first loaded, we run the elastic text plug-in in order to set the starting size of each of the text elements on the page.

Each time the window is resized, we'll run the function again to resize the text so it matches with the width of the window for best results.

```
001    /*
002        RESPONSIVE DESIGN WITH ELASTIC TEXT & IMAGES
003    */
004
005    (function($){
006        $.fn.elastictext = function(options) {
007
008        // Default options
009        var defaults = {
010            multiplier: 1.4,
011            lineheight: 1.4,
012            minWidth: 320,
013            maxWidth: 1200
014            };
015
016        // Assign the settings
017        var uoptions = $.extend(defaults, options);
018
019        return this.each(function(){
020            obj = $(this);
021            width = parseInt(obj.width());
022            if (width<uoptions.minWidth) width=uoptions.
minWidth;
023            if (width>uoptions.maxWidth) width=uoptions.
maxWidth;
024            fontSize = parseFloat(uoptions.
multiplier*(width/uoptions.minWidth))+"em";
025            obj.css({fontSize:fontSize,lineHeight:(parseFl
oat(fontSize)+(uoptions.lineheight-
parseFloat(fontSize)))+"em"});
026            });
027        }
028    })(jQuery);
029
030
031    $(document).ready(function(){
032
033        // Assign the starting sizes
034        $("article p.intro").
elastictext({multiplier:1.15});
035        $("article h2").elastictext({multiplier:1.8,linehei
ght:1});
036        $("header h1").elastictext({multiplier:3.05});
037
038        // Each time the window is resized, recalculate the
font sizes
039        $(window).resize(function(){
040            $("article p.intro").
elastictext({multiplier:1.15});
041            $("article h2").elastictext({multiplier:1.8,lineh
eight:1});
042            $("header h1").elastictext({multiplier:3.05});
043        });
044    });
```

Get users from A to B with Geolocation & Google Maps

Find out your users' location and use that data intelligently in your apps

tools | tech | trends Dreamweaver, Google Maps API
expert Mark Shufflebottom

On the CD
• Start folder
• Finished folder

<Above>
• Allow the browser access to your current location and the UK map quickly updates to dynamically plot your route, as shown above in both Map and Satellite views
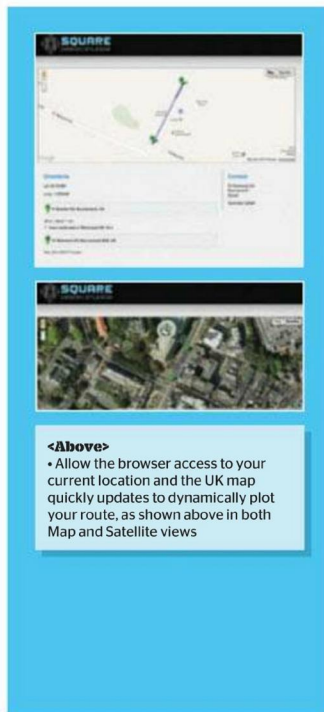
Working with HTML5 offers us as designers a suite of JavaScript APIs that go far beyond the semantic tags that are on offer in the actual HTML. Utilising these APIs grants us access to the geolocation data of latitude and longitude position, which is taken from location information sources such as GPS, IP address, RFID and Wi-Fi. Because the data comes from such devices there is no guarantee that the reading will always be accurate but most of the time it works pretty much spot on.

In this tutorial we are going to focus on intelligent driving directions to a fixed location such as an office. Why ask your user where they would like driving directions from if you can grab that information yourself by making the most of the Geolocation API? Obviously you should already know the location of the destination, so you can automatically provide the instructions for how the user can get there based on this information. This kind of intelligent web application is going to become even more useful in the future as we see further traffic to our sites from mobile content triggered from smartphones, whose users don't want to be inputting addresses on a tiny device. We will take advantage of the Google Maps API to display our directions and route directly on the page.

❝ Why ask your user where they would like driving directions from if you can grab that information with an API? ❞

01 Link to code libraries

From the cover CD, copy the Start folder to your hard drive and open 'index.html' in Dreamweaver. In order to create our application, we need to link to a couple of code libraries. The first is the obvious Google Maps API so that we can display custom maps and the second is jQuery. To kick off, add the following lines of code to the head section.

```
001 <script type="text/javascript" src="http://
maps.googleapis.com/maps/api/
js?sensor=false"></script>
002 <script src="http://ajax.googleapis.com/
ajax/libs/jquery/1.7.1/jquery.min.js"></script>
```

02 In the body section

Scroll down to the body section of the document and look for the closing </header> tag. Position your cursor just after this tag, press Return and add the div tag that's shown below. This tag will handle the Google map that we'll display via JavaScript code later, hence naming the ID as 'map'.

```
001 <div id="map"></div>
```

03 On the right track

Now position your cursor just after the opening article tag and before the opening aside tag that follows it. Hit Return and add the div tag as shown below. This tag is going to show the directions to the address later on from Google's API. Notice that we put a header tag in here as well so that the user understands the section.

```
001 <div id="directions"><h2>Directions</h2></
div>
```

04 Perfect style

Now scroll up to the head section of your document and find the closing </style> tag. Just before this tag add the following CSS code which will help to style the map section on the page. The map is made to be the width of the document which we want to be 960 pixels wide and 350 pixels high.

```
001 #map {
002     padding: 0;
003     height: 350px;
004     width: 960px;
005     margin-top: 0;
```

05 Continue styling the map

Add the remaining styling to the map that positions it in the centre of the page by having auto as the left and right margin. We also want to apply a dark grey solid line around the edge of the map so that it better stands out from the background.

```
001     margin-right: auto;
002     margin-bottom: 0;
003     margin-left: auto;
```
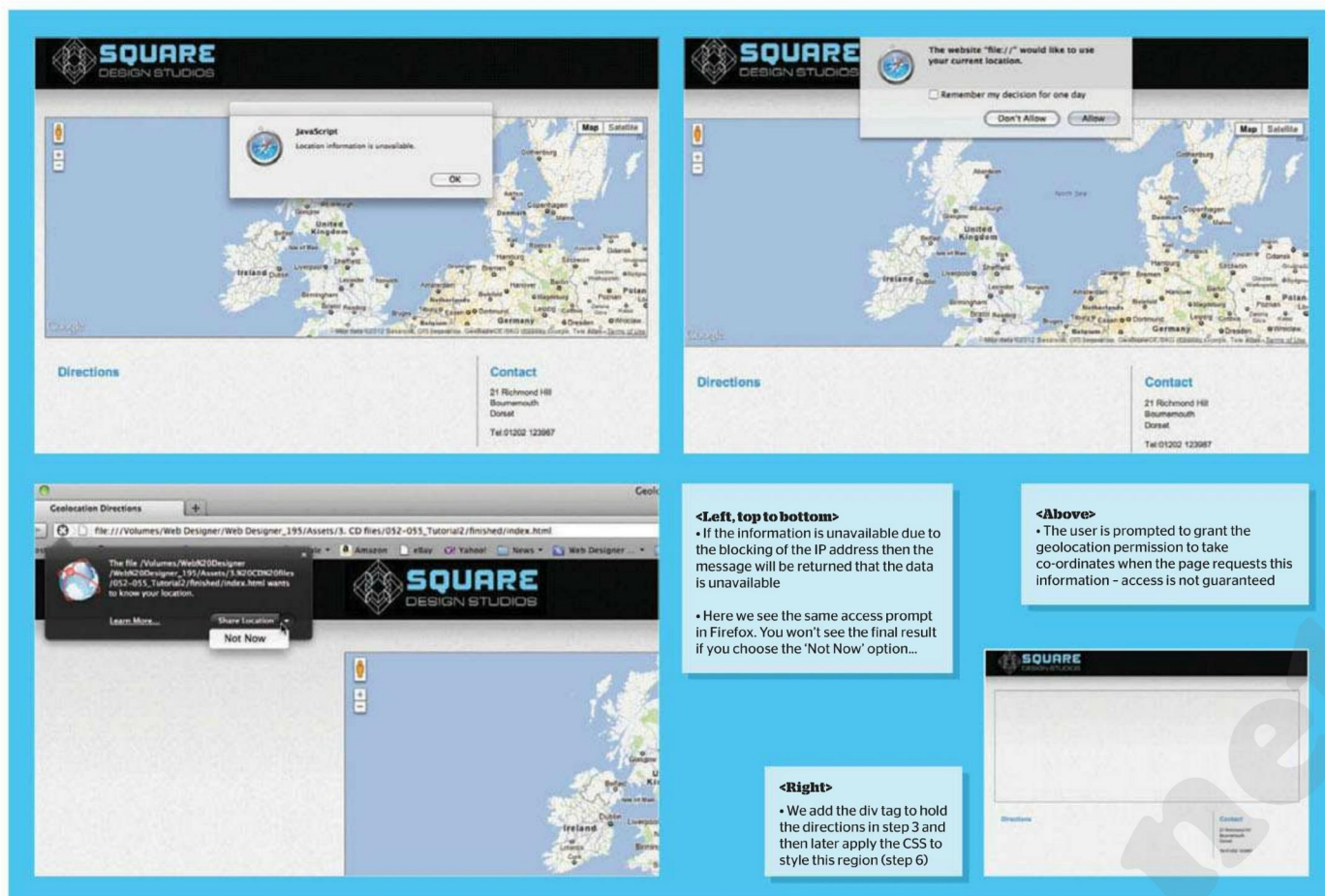
<Left, top to bottom>
• If the information is unavailable due to the blocking of the IP address then the message will be returned that the data is unavailable

• Here we see the same access prompt in Firefox. You won't see the final result if you choose the 'Not Now' option...

<Above>
• The user is prompted to grant the geolocation permission to take co-ordinates when the page requests this information - access is not guaranteed

<Right>
• We add the div tag to hold the directions in step 3 and then later apply the CSS to style this region (step 6)

```
004    border: 1px solid #666;
005 }
```

06 Directions style

Now we add the style for the directions. We float this div to the left and apply 20 pixels padding on the left-hand side to give it plenty of space. The contact information is floated to the right in the aside tag, hence why this is placed to the left. This is also set to be 650 pixels wide, giving plenty of space for the directions.

```
001 #directions {
002   float:left;
003   margin: 0;
004   padding-left: 20px;
005   width: 650px;
006 }
```

07 Add the JS code

After the closing </style> tag add the following JavaScript, which will be the start of our main code. We call a jQuery (document).ready function to check the document has loaded. Once it has, then we check to see if there is support for geolocation and, if there is, grab the current position.

```
001 <script>
002 $(document).ready(function () {
003 if (navigator && navigator.geolocation) {
004     navigator.geolocation.
getCurrentPosition(geo_success, geo_error);
```

08 Measure your success

Notice in the previous step we have geo_success and geo_error; these are functions that are called if you are successful or blocked in some way. Now we add an else statement that throws an alert if geolocation is not supported by the device that's rendering the page. Our remaining code will go before the closing </script> tag.

```
001     } else {
002         alert('Geolocation not supported.');
```

```
003     }
004 });
005 </script>
```

09 Grab the position

If we are successful in retrieving a geolocation then the geo_success function is called. Here we declare the geo_success function bringing the position in; this in turn calls the getLoc function. Notice that in the brackets we pass across the latitude and longitude co-ordinates so that we can use that information within the getLoc function.

```
001 function geo_success(position) {
002     getLoc(position.coords.latitude,
position.coords.longitude);
003 }
```

10 Declare any errors

Even if geolocation is supported, it doesn't always mean that we can get access to that information

<Above>
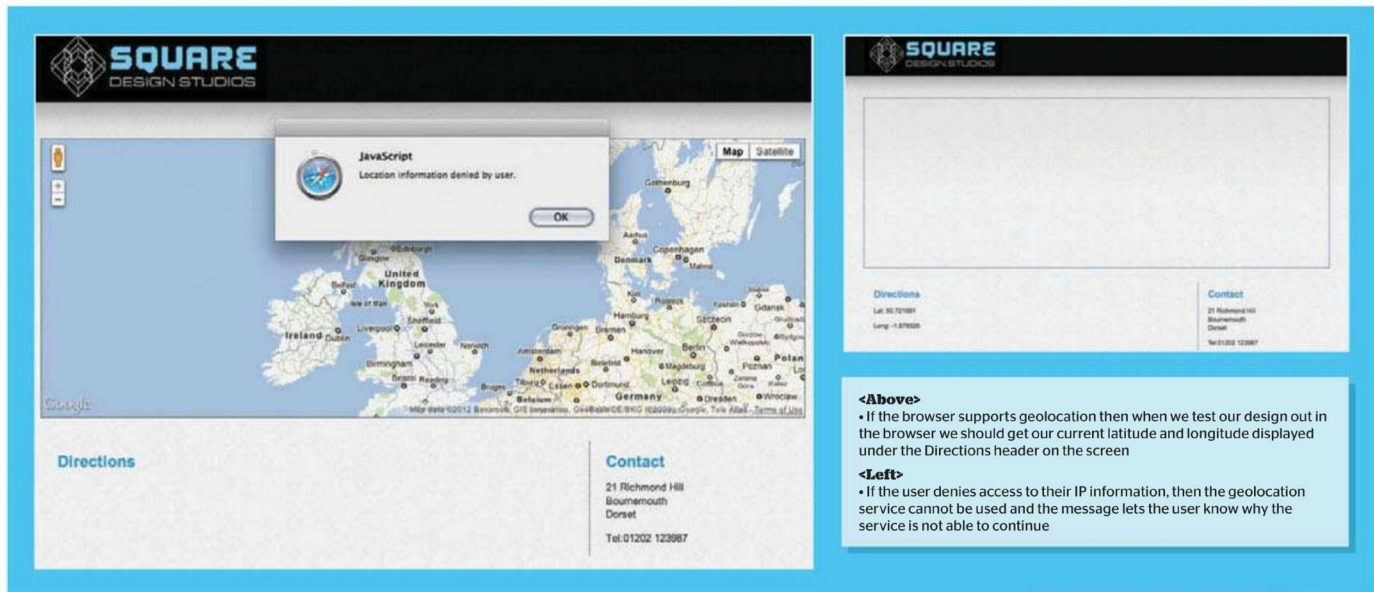• If the browser supports geolocation then when we test our design out in the browser we should get our current latitude and longitude displayed under the Directions header on the screen

<Left>
• If the user denies access to their IP information, then the geolocation service cannot be used and the message lets the user know why the service is not able to continue

and so, once a query for the geolocation is made, we feed back to the user if there is a problem. The first error is if the user denied access to the geolocation, and the second is if it's just not available.

```
001 function geo_error(err) {
002     if (err.code == 1) {
003         alert('Location information denied
by user.');
004     } else if (err.code == 2) {
005         alert('Location information is
unavailable.');
```

11 More errors

The third error is if it has taken too long to get the location information. The final else statement provides a catchall if there is some other error. For each error that occurs the alert function is called so an alert box will pop up for the user in the browser. It's better to give the user a concise message rather than just displaying a random error code.

```
001     } else if (err.code == 3) {
002         alert('Location information timed
out.');
003     } else {
004         alert('An unknown error
occurred.');
005     }
006 }
```

12 Display the location

Now we declare the getLoc function. This takes the two arguments of latitude and longitude into the

function. We use jQuery here to append the text and the values of latitude and longitude into the directions div tag. Save your work and give this a test in your browser now. If all has gone well you should see the result under the Directions heading.

```
001 function getLoc(lat, long) {
002     $('#directions').append('<p>Lat: ' + lat
+ '</p>');
003     $('#directions').append('<p>Long: ' +
long + '</p>');
004 }
```

13 Declare variables

Move your cursor up to the top of the script section and add the following three variables before the (document).ready function. These variables will hold a

<Above>
• Different maps such as satellite view can be loaded by default, but here we've stuck with the road map given that we are going to display driving instructions

Differing travel modes
In our tutorial we set up the directions for use with a car. However, you can also use walking and cycling as modes of transport.

reference to the map and two variables that will render and call the actual directions from the Google Maps API into our web application.

```
001 var map;
002 var directionRenderer;
003 var directionsService = new google.maps.
DirectionsService();
```

14 Initial map display

Move your cursor inside the (document).ready function and add the code shown on the first line. We want an initial Google map to display while we access our user's geolocation. This variable stores a position that, when zoomed out, will be able to show the whole of the UK in the map display.

```
001 var loc = new google.maps.LatLng(55, -1.5);
```

15 Define the map

Add the next piece of code directly under the last. This variable of options sets the zoom level of the map to 5. It centres the map on our loc variable from the previous step. We then set the type of map that we want to display to Roadmap, which will clearly show the route for those travelling by car.

Get users from A to B with Geolocation & Google Maps

```
001    var myOptions = {
002       zoom: 5,
003    center: loc,
004    mapTypeId: google.maps.MapTypeId.ROADMAP
005
006       }
```

16 Display the map

The next line of code should be added under the last and places the map into the map div tag using our options from the previous step to set the map. Save your work and test this in your web browser; you will now see that your map is currently hovered over the UK ready to accept driving directions.

```
001 map = new google.maps.Map(document.
getElementById("map"), myOptions);
```
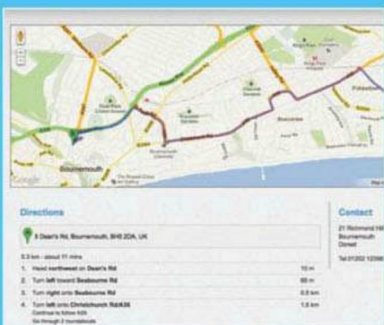
17 Start the directions

Add the next lines under the code from step 16. This JavaScript tells the application to start a new Google map direction lookup. It instructs the service to update the visuals of the map in the map variable from the previous step. It also tells it where to place the textual driving directions, which in this example will be positioned under the map itself.

```
001 directionsRenderer = new google.maps.
DirectionsRenderer();
002 directionsRenderer.setMap(map);
003 directionsRenderer.setPanel(document.
getElementById("directions"));
```

18 Start and end positions

Now move your cursor down to the getLoc function and comment out the two lines of code inside it using two forward slashes (//) at the start of each line. The start position is dynamically grabbed from the user's browser. The end position is the physical address of the destination you'd like to get them to.

<Above>
• Once the directions have been retrieved the display is updated with the journey on the map and the step-by-step directions are displayed below

```
001 var startPos = new google.maps.LatLng(lat,
long);
002 var endPos = new google.maps.
LatLng(50.7220, -1.8780);
```

19 Direction request

Directly under the previous code, add the following block of code. This request to the Google Maps API requires three arguments: the origin of the route which we grab from our startPos variable; the destination of the directions, again from our end position; and, finally, the travel mode, which in this instance we've set to Driving.

```
001 var request = {
002    origin: startPos,
003    destination: endPos,
004    travelMode: google.maps.
DirectionsTravelMode.DRIVING
005    };
```

20 Request the route

We now add the following code, which requests the route of the journey. It uses the request options from the previous step and returns the value in the variable result. The status tells us what is happening, so we use an if statement to check this is okay; if it is, we render the result out to the document.

```
001 directionsService.route(request, function
(result, status) {
002    if (status == google.maps.
DirectionsStatus.OK) {
003       directionsRenderer.
setDirections(result);
```

21 Just in case...

Of course there are a whole bunch of reasons why this might go wrong, so we need to keep our user informed of what is happening. As done previously, we make sure that we send a pop-up alert to the browser that will tell the user that the directions have failed. This lets them know what's going on, rather than being left out of the loop and confused as to how to continue with their journey, which would be frustrating.

```
001    } else {
002       alert("Directions failed: " +
status);
003       }
004    });
```

22 Save and test

We have now finished the code so make sure to save your work and refresh the page in your web browser of choice. Notice that when the directions load, the Google map automatically zooms to the extents of the directions, which is handy to show the full route ahead for the user. The directions should also be displayed as a list-type itinerary below the map.

Get Lat Lon

If you need to find the latitude and longitude of an address try this handy site: **www.getlatlon.com** – then query your directions with the Google Maps API.
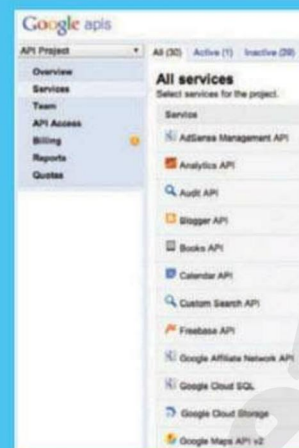
Working with the Google Maps API

To get started with the Google Maps API in your application, you really need to obtain an application key. To do this visit the API console at **https://code.google.com/apis/console** where you will need to sign in with a Google ID. At this point you will be provided with a list of services that Google offers. Selecting the Google Maps option will give you a key that allows you to use this service, but your app may be able to make use of other services such as Translate, Calendar or Analytics, so select as necessary. You then need to visit the API Access tab where you are prompted to create an OAuth 2.0 client ID. This requires you to name your product and upload a product logo. From here you will be granted your key which you insert into your webpage as follows:

```
001 <script type="text/javascript"
src="http://maps.googleapis.com/
maps/api/js?
002 key=YOUR_API_KEY&sensor=SET_TO_
TRUE_OR_FALSE">
```

How to make a WordPress theme adapt to any screen

Hate it when you find a WP theme you love, but then realise it looks awful on a smartphone? We fix that problem here

tools | tech | trends WordPress **expert** Thord Daniel Hedengren

On the CD
• N/A

Mobile devices are eating away at the visitor logs for websites all over the internet. Sure, most sites will look decent enough on your smartphone, but it's a nuisance to have to pinch and zoom to actually read something, right? This means mobile iterations of websites are more important than ever, and obviously this is something you don't want to miss out on.

There are many ways to make your website mobile accessible. If we look at WordPress, you'll see there are several plug-ins serving mobile-friendly versions of your site. Not everybody wants to use a plug-in though.

In this tutorial we use media queries to make the Twenty Ten theme responsive. That is to say, if the visitor is viewing via a small screen we'll make the theme adapt. Since we don't want to ruin the Twenty Ten template itself with our edits, we'll create a child theme to keep additions separate. Let's get started!

01 Make a child theme

First of all we need to create our child theme. As implied, child themes rely on parent themes which means that everything that doesn't appear in the child will be loaded from the parent. A child theme sits in its own folder in wp-content/themes/ and is activated just like any other theme. First, create a folder called 'twentyten-responsive' and, within a file called 'style.css', insert the following code:

```
001  /*
002  Theme Name: Twenty Ten Responsive
003  Theme URI: http://tdh.me/wordpress/twenty-
     ten-responsive/
004  Description: A child theme that makes
     Twenty Ten responsive.
005  Author: Thord Daniel Hedengren
006  Author URI: http://tdh.me/
007  Template: twentyten
008  */
009
010  @import url('../twentyten/style.css');
011
```

02 Activate the child theme

Use a development install of WordPress that has the Twenty Ten theme, which is shipped with WordPress by default but can also be downloaded from **wordpress.org/extend/themes/twentyten**. Upload the twentyten-responsive folder to wp-content/themes/ and activate the Twenty Ten Responsive theme from within WordPress. You'll notice that the site looks just like Twenty Ten, since at this point we haven't done anything but import Twenty Ten's stylesheet.

<Above>
• This is a bit easier to read on a mobile screen, but it's not exactly very pretty yet...

<Right>
• It's amazing what a little padding can do, but don't overdo it; make sure the amount works with the screen width

Menu matters

Menus for extensive sites are not well-suited to the small screen. Perhaps you should have two different ones, showing and hiding them with CSS? It might fit your project better than just scaling them.

03 ### We need header.php too

Before we can do any fun stuff with media queries in style.css we first need to add a tiny bit of code to header.php. Touching Twenty Ten itself is a big no-no, so copy the header.php file from the wp-content/themes/twentyten/ folder and place it in the twentyten-responsive folder. In the meta section of our own header.php, around line 14, add the following line which will let us play with the viewport:

```
001 <meta name="viewport" content="initial-
scale = 1,user-scalable=no,maximum-scale=1.0"
/>
002
```

04 ### Decide on the media queries

In this tutorial we're not going for a fully fluid version of Twenty Ten but rather adapting the layout for various set widths. The default site width is 980px, which most modern displays will be able to handle. We'll have a media query for 320px width and up, and then we'll add some additional rules for 480px and up.

The next step is 640px, and then we'll go over the 980px width rules as well (ie the default width).

05 ### Set up the media queries

With our plan decided, let's start to write the media queries. This is done in the child theme we set up in step 1 since all edits should be kept free from the parent theme. Open style.css from the twentyten-responsive folder and add the following queries. Do read the commenting within the code for a quick insight as to what each media query is meant to do.

```
001 /* ===============
002    MEDIA QUERIES */
003
004 /* 320 px and up */
005 @media only screen and (min-width: 320px) {
006
007    /* This goes for everything 320 pixels
and up */
008
009 }
010
011 /* Up to 480 px */
012 @media only screen and (max-width: 480px) {
013
014    /* Up to 480 pixels */
015
016 }
```

```
017
018 /* From 481 px and upwards */
019 @media only screen and (min-width: 481px) {
020
021    /* For screens 481 pixels and wider */
022
023 }
024
025 /* From 640 px */
026 @media only screen and (min-width: 640px) {
027
028    /* Special styling from 640 pixels and
up */
029
030 }
031
032 /* 980 px and up */
033 @media only screen and (min-width: 980px) {
034
035    /* Where we set everything straight again
*/
036
037 }
```

06 ### Decide what to hide

The key to getting the versions for smaller screens to both look good and work well is to make sure you hide unnecessary elements, and find new places for essential ones. In this example we'll hide the header on top up until the desktop version, and we'll float the right-hand column to the bottom of the page. Sometimes you'll be better off with other priorities, but deciding which elements are crucial is key so make sure you think things through before touching any code.

07 ### Mobile first

There is a reason there is so much talk about 'mobile first'. The first media query – 320 pixels and up – will basically strip away everything we don't want or need from Twenty Ten's layout, and reposition other things. These rules will apply at min-width: 320px, which is the standard for smartphones, so we'll have to make some more tweaks to our edits later. For now, let's get rid of some stuff from Twenty Ten that isn't suitable for small screens. You'll find these referenced styles in Twenty Ten's style.css, which we must import to the top of our child theme's style.css file.

```
001 /* 320 px and up */
002 @media only screen and (min-width: 320px) {
003
004    #wrapper {
005       margin: 0;
006       padding: 0; }
007
008    #site-title {
009       width: 100%;
010       margin-bottom: 0;
011       text-align: center; }
012
```

<Above>
• Our template shown reformatted for tablet, shown here in landscape aspect

<Left>
• Our template shown in default desktop format, how you'd expect to encounter it fullscreen

<Right>
• Our template adapted for the slimmer width of a smartphone screen such as iPhone, conventionally shown in portrait aspect
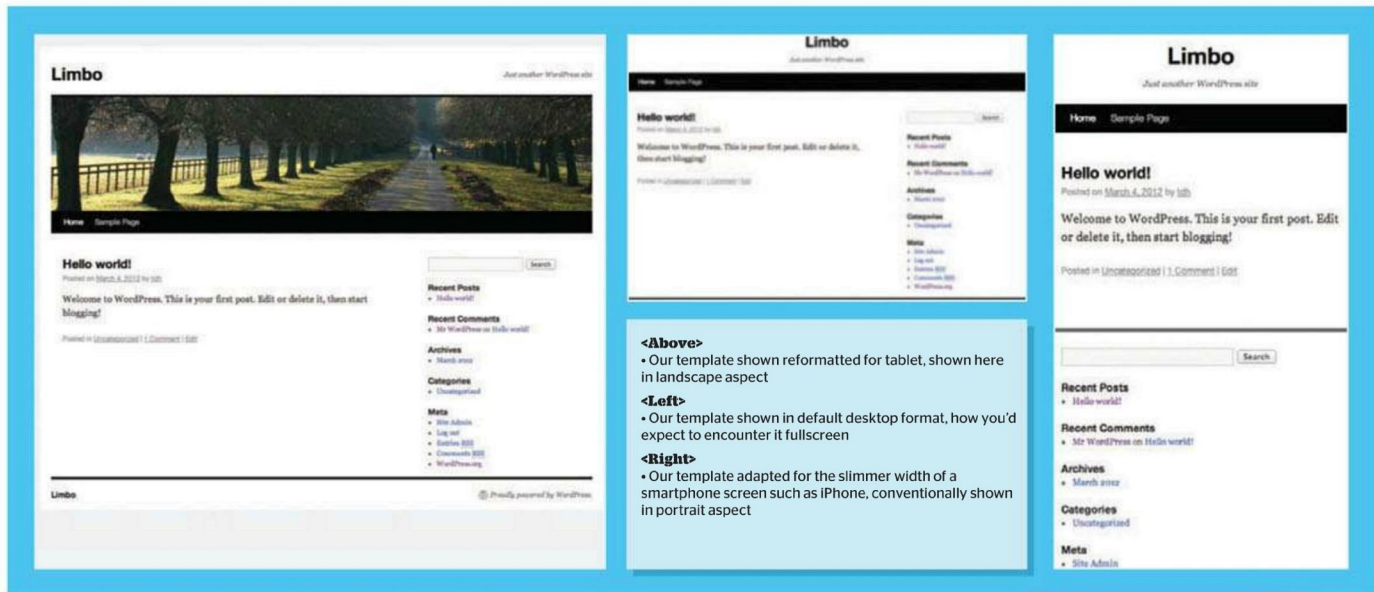
Consider widget content

Depending on the sort of widget content you have, you may have to make changes to the stylesheet or hide certain widgets altogether.

```
013    #site-description {
014        width: 100%;
015        float: left;
016        margin-top: 10px;
017        text-align: center; }
018    #access, #access .menu-header, div.menu,
#colophon, #branding, #main, #wrapper {
019    @media only screen and (min-width: 481px) {
020        width: 100%; }
021
022    #branding img {
023        display: none; }
024
025    div#content {
026        width: 100%;
027        margin: 0; }
028
029    div#primary {
030        width: 100%;
031        margin-top: 20px;
032        padding-top: 20px;
033        border-top: 3px double #000; }
034
035    #main .widget-area ul {
036        padding-right: 0; }
037
038    #footer {
```

```
039        margin-bottom: 0; }
040
041    #site-info, #site-generator {
042        width: 100%;
043        float: left;
044        text-align: center; }
045
046 }
```

08 Make it pretty for small screens

While this simple piece of code (which basically just overwrites the original styles from the Twenty Ten theme) will make the site infinitely more readable when viewed on mobile devices, it is in need of a good deal of beautifying by us. Let's add some spacing in the next media query. The reason we're doing it here and not on top is that we'll have different amounts of space later, so it's not as global as the above changes.

```
001  /* Up to 480 px */
002  @media only screen and (max-width: 480px) {
003
004      .post, div.page, li.widget-container {
005          padding: 0 10px; }
006
007  }
008
```

09 Make it pretty for slightly larger screens

Screens that are larger than 480px but smaller than 640px might benefit from a little more spacing, so let's increase the padding we just set for the up-to-480px screens in the previous step, from 10px to 15px.

```
001  /* From 481 px and upwards */
002  @media only screen and (min-width: 481px) {
003
004      .post, div.page, li.widget-container,
#comments {
005          padding: 0 15px; }
006
007  }
```

10 Reposition the sidebar

You might wonder why we have a media query from 640px and up? The reason is that from this size it works with the sidebar positioned on the right-hand side. Our problem is, we need it to scale appropriately, so the width will be listed in the form of a percentage.

```
001  /* From 640 px */
002  @media only screen and (min-width: 640px) {
```

66 Galleries need a bit of loving to look good on smaller screens too 99

```
003
004    #container {
005        width: 62.5%;
006        margin: 0; }
007
008    .post, div.page, li.widget-container {
009        padding: 0 20px; }
010
011    div#primary {
012        width: 32.5%;
013        margin-top: 0;
014        padding-top: 0;
015        border: 0; }
016
017 }
```

11 Back to normal

Right now, the desktop version of the site looks pretty bad, much like the screenshots shown across the page. This is normal – we have overwritten a lot of styles after all! In our final media query – the one for 980px and up – we'll set things straight again, basically reverting to the standard Twenty Ten.

```
001 /* 980 px and up */
002 @media only screen and (min-width: 980px)
{
003
004    #access .menu-header, div.menu,
#colophon, #branding, #main, #wrapper {
005        width: 940px; }
006
```


<Above>
• Watch embedded videos on screens ranging from 320px to 980px thanks to our now responsive theme

```
007 #wrapper {
008    margin: 20px auto;
009    padding: 0 20px; }
010
011 #site-title {
012    width: 700px;
013    margin-bottom: 18px;
014    float: left;
015    text-align: left; }
016
017 #site-description {
018    width: 220px;
019    float: right;
020    text-align: right; }
021
022 #branding img {
023    display: block; }
024
025 #footer {
026    margin-bottom: 20px; }
027
028 #site-info {
029    width: 700px;
030    float: left;
031    text-align: left; }
032
033 #site-generator {
034    width: 220px;
035    float: right;
036    text-align: right; }
037
038 }
```

12 What we've got so far…

At this stage we actually have a pretty decent mobile version of the site, along with a middleground for mid-sized, and the standard setup for desktop. You would think we could stop here, and if you only used text that would be the case, but alas, images and other things need to behave as well. We haven't looked at single posts and pages either, so there's still work to do.

13 Single posts and pages

A quick look at a single post tells us that the comments need styling; at the moment, they hit the outer margins of the window. Luckily this is easy enough to take care of, since the comments section sits in div#comments. Remember the padding code introduced in step 7, and added to in steps 8 and 9? Find that, and add #comments to it and you'll be fine. The code from step 7 would look like this:

```
001 /* Up to 480 px */
002 @media only screen and (max-width: 480px) {
003
004    .post, div.page, li.widget-container,
#comments {
005        padding: 0 10px; }
006
007 }
```

14 Getting the navigation links right

It's not just the comments code that needs to be positioned, as other elements will show up when you're making an existing theme responsive as well. In Twenty Ten, one such element is the navigational links, linking to the previous page in archives as well as to the previous post on single posts. We only need to worry about these when #wrapper is set to no padding, so just add the .navigation class to the same media queries as you did in step 13, like this:

```
001 /* Up to 480 px */
002 @media only screen and (max-width: 480px) {
003
004    .post, div.page, li.widget-container,
#comments, .navigation {
```

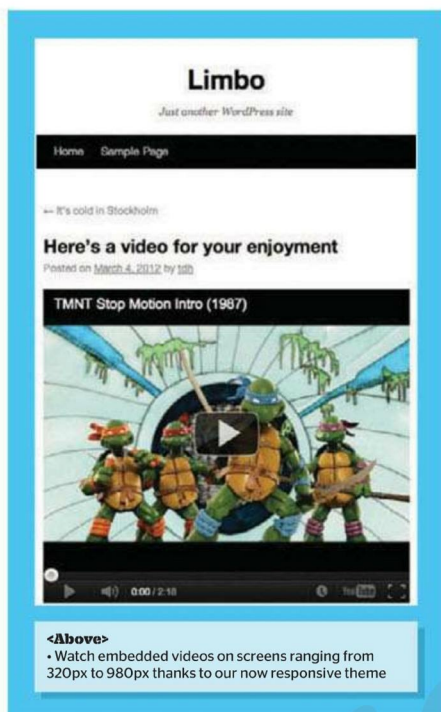

The Internet Explorer solution for your HTML5 needs

Twenty Ten isn't coded in HTML5 so you won't find this problem here, but more and more WordPress themes are. The thing is, Internet Explorer just doesn't play nicely with HTML5 and CSS3 all the time. You'll be fine from version 9 and up, but prior to that you'll have problems. There are numerous boilerplates that solve this in a more or less elegant fashion, but the best solution is perhaps Remy Sharp's excellent JavaScript HTML5 shim, which is hosted for free at Google Code. It will make the HTML5 tags behave as expected in IE6-8. You can include it in any project, using the IE conditional tag:

```
001 <!--[if lt IE 9]>
002 <script src="http://
html5shim.googlecode.com/svn/
trunk/html5.js"></script>
003 <![endif]-->
```
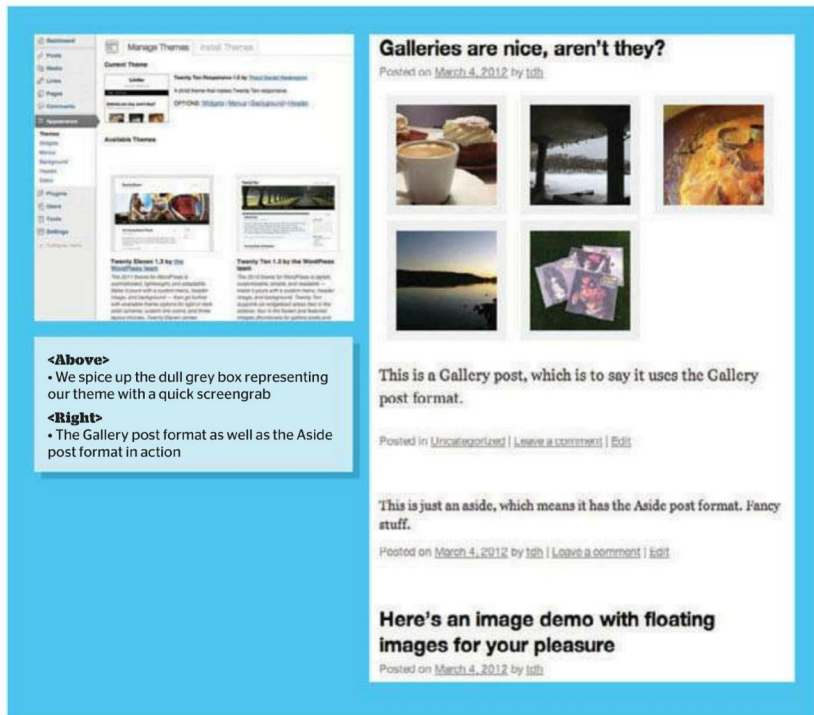
It works perfectly well with WordPress themes as well, although you should consider using wp_enqueue_script() and possibly wp_register_script() to queue the HTML5 shim script with your theme.

<Above>
• We spice up the dull grey box representing our theme with a quick screengrab

<Right>
• The Gallery post format as well as the Aside post format in action

```
001 div.menu {
002     font-size: 11px; }
003
004 #access a {
005     padding: 0 8px;
006     line-height: 32px; }
007
```

18 Aside posts

Twenty Ten supports the Aside post format, and by default so does our responsive child theme. We need to make sure that these posts adhere to the new margin positioning as well, which unfortunately isn't as easy as just adding .format-aside to the padding blocks introduced in step 7, due to specific Twenty Ten styling for asides. Luckily we can use margins instead, so add this code to the up-to-480px media query:

```
001 .format-aside {
002     margin-left: 10px;
003     margin-right: 10px; }
```

Then we'll apply 15px instead of 10px to the 481px-and-up media query, and 20px for the 640px-and-up version, all corresponding to their respective padding values, of course. It's as simple as that.

19 Gallery tweaks

Galleries need a bit of loving to look good on smaller screens too. The easiest solution is to change the size of img.attachment-thumbnail on the media queries, starting with the one up to 480px.

```
001 img.attachment-thumbnail {
002     width: 75%;
003     height: auto; }
```

For 481px and upwards, make it slightly bigger:

```
001 img.attachment-thumbnail {
002     width: 80%;
003     height: auto; }
```

Then up in size again for 640px and above:

```
001 img.attachment-thumbnail {
002     width: 75%;
003     height: auto; }
```

Lastly, revert to default for 980px and over:

```
001 img.alignleft, img.alignright {
002     width: auto;
003     height: auto; }
```

```
005     padding: 0 10px; }
006
007 }
```

15 Dealing with embedded video content

Embeds are something that you'll almost certainly have to handle. After all, when they are embedded into your width they'll be in a set width and most likely that's a wee bit more than the 320px width of an iPhone, for example. For embedded video this is simple enough; just by adding the following code to our first media query – the one for 320px and up – YouTube videos and their like will be optimally presented:

```
001 embed, iframe, object {
002     width: 100%; }
```

16 Scale your imagery

By default the Twenty Ten theme already scales images so that they won't be wider than the content column. That means that we won't have to fiddle with them, but in other cases you'd want to add a max-width to .aligncenter and .alignnone most likely. What Twenty Ten won't take care of is images floating to the left or right, which will have to be scaled down to make sense on smaller screens.

First there's the up-to-480px resolution, where we'll scale the images to 35%:

```
001 img.alignleft, img.alignright {
002     width: 35%;
003     height: auto; }
```

Then there's 481px and upwards:

```
001 img.alignleft, img.alignright {
002     width: 40%;
003     height: auto; }
```

Next, there's 640px and up:

```
001 img.alignleft, img.alignright {
002     width: 45%;
003     height: auto; }
```

Finally, let's reset the whole thing in the 980px-and-upwards media query:

```
001 img.alignleft, img.alignright {
002     width: auto;
003     height: auto; }
```

17 Tighten up the menu

Another object that will need some attention is the menu, as it won't fit too many things per row on a smartphone. Since this is mostly an issue on really small screens, let's just reduce the font size and the spacing a bit, as well as the line height. If you have an extensive menu you might have to create a completely different solution altogether. Adding this to the up-to-480px media query should do the trick:

20 Add screenshot.png

Finally, the grey box representing the Twenty Ten Responsive theme on the Themes page within WordPress looks a bit bland at the moment. Let's screengrab the theme when it is smaller and create a 300 x 225px PNG from the shot. Name this 'screenshot. png' and add it to the twentyten-responsive folder. The preview box on the Themes page now looks a lot more appealing, we're sure you'll agree.

heart internet

100% UK Based

UK's MOST ADVANCED RESELLER HOSTING

Sign up to our award winning Reseller Hosting and sell unlimited websites with unlimited web space, bandwidth, databases, email and more, all at **no extra cost.**

FIRST MONTH ONLY £1

The complete Reseller Hosting solution

Sell unlimited websites and domain names, as well as our wide range of additional products to fit your customer's needs. Use our hosting billing & CRM solution 'HostPay' to sell in real time.

100% YOUR brand

Every aspect of your hosting is white label and can be branded as your own, including your customer's control panel, webmail, welcome emails and anything else you can think of!

Create your own packages

Customise each account to any level you like. From how much web space and bandwidth your customers get, to selecting what web apps they can install from their control panel.

24/7 UK support

We don't employ any contractors, temps or external support services; all our support team are 100% based in our UK offices and full time members of staff.

No hidden fees

All the features you need to attract and manage customers are included as standard. Unlike our competitors we don't charge you for basic functionality, such as your customers getting a brandable web hosting control panel or additional MySQL databases.

Increase your profits with exclusive reseller discounts

25% OFF
- Dedicated Servers
- Virtual Servers
- goMobi
- SSL Certificates

50% OFF
- Hosted Exchange
- Premium Email

30% OFF
- Bulk Domain Names

> " Heart Internet support their customers with market-leading hosting technology at great prices "
>
> **Suzi Perry**
> Technology TV Presenter

As well as selling unlimited websites, pick and choose from a wide range of additional products, including:

Sell Virtual Servers
Only £8.99 per month

BRAND NEW

EXCLUSIVE 25% OFF

- Create your own VPS packages
- Full API integration
- Linux & Windows
- Dell & Intel hardware
- Full white label control panel for your customers
- Sell VPS in real time

Sell Dedicated Servers
Only £59.99 per month

EXCLUSIVE 25% OFF

- No annual contract
- Linux & Windows
- Dell & Intel hardware
- No set up costs
- Full white label server control panel for your customers
- Sell servers in real time

Sell Hosted Exchange
Only £4.99 per mailbox

EXCLUSIVE 50% OFF

- MS Exchange 2010 as standard
- Free Outlook 2010 for you & your customers
- 10GB storage
- Transfer in Exchange 2003 & 2007 easily
- Sell Exchange in real time

Find out more: www.heartinternet.co.uk
Call us: 0845 644 7750
Normal Price from only £29.99 per month

heart internet
www.heartinternet.co.uk
Web hosting | Reseller hosting | VPS | Servers

In-line Twitter feed tooltips with jQuery

Get one-click, subject-specific Twitter updates in your browser using this nice and easy jQuery technique

tools | tech | trends Dreamweaver (or editor of choice)
expert Steven Jenkins

On the CD
• Twitter tooltip folder

‹Above›
• The jQuery Twitter Search Plugin provides all the back-end code for displaying tweets that you need

witter is used by millions worldwide and provides a great platform for the latest opinions and news on almost any subject. However, by its very nature Twitter is awash with white noise, and filtering out the unwanted or irrelevant news is a must. This clever jQuery technique uses a search term to only display the tweets related to a keyword.

This tutorial looks at users clicking specified text anywhere within the confines of a page and producing a pop-up tooltip with tweets related to the chosen text. For example, choose 'CSS3' and all the latest tweets containing CSS3 are neatly stacked and displayed in the tooltip. As the technique employs a search term to elect which tweets to show, the technique can be applied to as many terms within the text of a page as is needed.

This tutorial uses a selection of jQuery plug-ins found at the official jQuery (**http://jquery.com**) and jQuery (**http://jqueryui.com**) sites. The accompanying code can be found at **http://bit.ly/dCvYlq**; this provides the elements you will be needing to get started with creating the Twitter tooltip. Here we show you how to implement the jQuery and manipulate the code and styling to create a clever, imaginative and personalised option for displaying subject-specific tweets.

01 Add jQuery

First create a folder to store all the files needed for this tutorial. Add the 'index' file found on the disc or a HTML page where you wish to add the tooltip. We have included the HTML and CSS in the same page for easy reference. Now open the 'index' file and add the jQuery references into the head of the page.

```
001 <script type="text/javascript" src="http://
ajax.googleapis.com/ajax/libs/jquery/1.4.2/
jquery.min.js"></script>
002 <script type="text/javascript" src="http://
ajax.googleapis.com/ajax/libs/jqueryui/1.7.2/
jquery-ui.min.js"></script>
```

02 Add files

Now add the 'Twitterpop.js' and 'Twitter.search.js' files, downloaded from the Tympanus link in the intro, to the folder where the index file has been stored. Next, include the reference to the two files directly under the jQuery references in the previous steps and save the page. If stored in a different location remember to add the correct path, eg 'scripts/ jquery.Twitterpopup.js'.

```
001 <script type="text/javascript" src="jquery.
Twitterpopup.js"></script>
002 <script type="text/javascript" src="jquery.
Twitter.search.js"></script>
```

03 Apply the JS

Now add the JavaScript code needed to get the tooltip working. Place this underneath the other reference scripts. This code will look for the named div tag, do a search and then open a pop-up window with the results of the selected name. In the example code the div tag is called #wrapper; change this to refer to the div tag in your HTML page if necessary.

```
001 <script>
002 $(function() {
003 $('#wrapper').find('.Twitter_search').
Twitterpopup();
004 });
005 </script>
```

04 Create class

The pop-up tooltip needs an activation point to call it into action. This is a link with an id of Twitter_search. The class can be directly added to the CSS (see below) without any formatting and then applied via the Class drop-down list in the Properties window if using Dreamweaver, or manually if hand-coding.

```
001 .Twitter_search{
002 }
```

05 Style class

To emphasise that a link relates to a Twitter tooltip a visual pointer is going to be added to help identify it – ideally this would be a Twitter 't'. Head to

> **By its very nature Twitter is awash with white noise, and filtering out the unwanted or irrelevant news is a must**

<Above>
• The jQuery library can be found at **http://jquery.com**. Get the latest version to produce contemporary code

twitter.com/about/resources/logos to get an official logo and resize to around 15 x 15px. This can vary depending on the size of the body in the page used.

06 Style link

The Twitter icon just created is to be added as a non-repeating background image and placed to the right. To change to left simply swap right to left in the background selector. The padding should be around 20px. Finally, a cursor is added to enhance the link.

```
001 .Twitter_search{
002 padding-right:20px;
003 background: url(images/Twitterbird15.png)
no-repeat right center;
004 cursor:pointer;}
```

07 Apply the link

The Twitter_search class is now ready to be applied to any of the text within the current HTML page. Select the desired text and apply Twitter_search from the Class drop-down list in the Properties window (if using Dreamweaver). Alternatively, hand code the class to the desired text and make it bolder for emphasis.

```
001 <span class="Twitter_search"><strong>CSS3</strong></span>
```

To bold up the text add the following:

```
001 <span class="Twitter_search"><strong>CSS3</strong></span>
```

08 Test the tooltip

The initial setup is almost complete, so save the page and preview in your preferred browser. You will notice that the tooltip is not styled properly, spreading itself across the whole page. To style the tooltip the code below will need to be added to the CSS. Modify the width to suit.

```
001 .search_results {
002 width: 300px;
003 position:absolute;
004 display:none;}
```


<Above>
• The jQuery UI library brings together a whole host of effects for the user interface

09 jQuery UI code

Currently the Twitter tooltip does not have a close button and cannot be resized. The inclusion of the jQuery UI is next on the agenda. These are included in the Tympanus download. Add these to the same location as the index file or place into a specific folder (remember to correct the path accordingly).

```
001 <link type="text/css" href="jquery.ui.theme.css" rel="stylesheet" />
002 <link type="text/css" href="jquery.ui.core.css" rel="stylesheet" />
003 <link type="text/css" href="jquery.ui.resizable.css" rel="stylesheet" />
```

10 Make it resizable

The introduction of the jQuery UI reference will now make the Twitter tooltip resizable; give it a try! To make it more obvious an icon needs to be applied. Create this in Photoshop, or your image editor of choice. Make it around 11 x 11px and then add as a background to the .TwitterSearchContainer class.

```
001 .TwitterSearchContainer{
002 background:#aaa url(images/resize.png) no-repeat right bottom;
003 }
```

11 Add a close button

The final addition to the tooltip is a close button. This is contained in the TwitterSearchClose class. Once again create a close icon in Photoshop and add this into the CSS of the current page. This will appear in the top-right corner of the tooltip, as you'd expect. Modify the background colour and reposition if you wish.

```
001 .TwitterSearchClose{
002 position:absolute;
003 right:0px;xx
004 top:0px;
005 background:#CCC url(close.png) no-repeat center right;
006 width:13px;
007 height:13px;
008 cursor:pointer;
009 }
```

12 Locate tooltip CSS

The tooltip will now operate in its default state; the width of the tooltip was determined in step 8. However, a number of elements can be styled to suit. Open the Twitter.search.js and head to line 176 to locate the default CSS applied to the tooltip. Copy and paste this into the editor to avoid making any mistakes.

```
001 a:     { textDecoration: 'none', color: '#1485A5' },
002 bird: { width: '50px', height: '20px', position: 'absolute', left: '-30px', top: '-20px', border: 'none' }
```
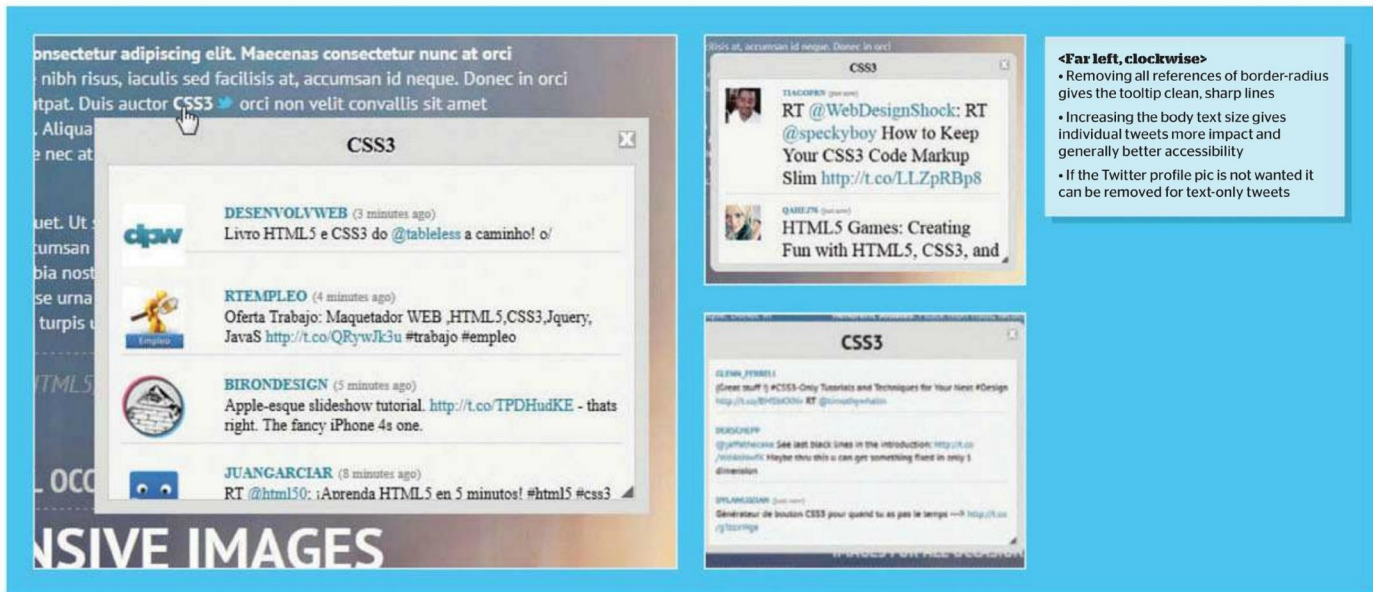
13 Container

The container is the area within the frame where all the tweets are inserted. At this point, this is assigned a background colour, a fixed height and a border. Leave overflow as hidden and adjust the other assets as you desire - for example, tweak the height, but the frame will need to be modified to match.

```
001 container: {
002 overflow: 'hidden', backgroundColor:
'#eee', height: '253px', border : '1px solid
#ccc'}
```

14 Tooltip frame

The tweets are contained within a frame which has a number of its elements predefined. The height can be adjusted to suit, but if the container is not adjusted as well, you will be left with a gap. Adjust the border; the top is liquid and you'll find it will accommodate the text title. Adjust the border radius or remove altogether to get sharp squared-off edges.

```
001 X frame: { height: '280px', border: '10px
solid #C2CFF1', borderRadius: '10px', '-moz-
border-radius': '10px', '-webkit-border-
radius': '10px', '-moz-box-shadow': '1px 1px
5px #000', '-webkit-box-shadow': '1px 1px 5px
#000' },
```

15 Twitter text

The text styling refers to the text within a tweet. The size of the text can be modified, eg to 25px to show one large-text tweet at a time. The border creates the separator between tweets - you can modify colour and/

or size, while the padding creates a decent amount of space at the bottom of a tweet to keep things clean.

```
001 text: { fontSize: '11px',
borderBottom:'1px solid #ddd',
paddingBottom:'6px' },
```

16 Tooltip title

The title refers to the keyword that is chosen within the text – ie the one you are wanting to search the Twitterverse for. This is the title that is added to the tooltip. The main attributes you will need to modify here are the text size and alignment. Increasing the text size will push down the container, so the frame will also need to be tweaked accordingly.

```
001 title: { cursor: 'move', backgroundColor:
'#C2CFF1', margin: 0, padding: '0 0 5px 0',
textAlign: 'center', fontWeight: 'bold',
fontSize: '16px', textShadow: '1px 1px 1px
#fff',position: 'relative' }
```

17 Twitter profile

By default, a Twitter profile picture is shown along with every tweet. This has a default size of 48 x 48px, the standard image size used on Twitter. To resize the image, all you have to do is adjust the height and width elements. To place the image to the right simply change the parameter to right.

```
001 img: { 'float': 'left', margin: '5px
10px 5px 0px', width: '96px', height: '96px','-
moz-box-shadow': '1px 1px 5px #000', '-webkit-
box-shadow': '1px 1px 5px #000'}
```

18 Search defaults

Sitting above the default styling are the Twitter search defaults. With some simple mods a few elements can be changed. To remove the profile picture and have a text-only tweet, for instance, just set avatar to false. Alternatively, to modify the time between tweet scrolls adjust timeout; add a larger number for a longer delay and a smaller number for a shorter one.

```
001 avatar: true,
002 bird: true,
003 birdLink: false,
004 birdSrc: 'http://cloud.github.com/
downloads/malsup/Twitter/tweet.gif',
005 timeout: 4000,
```

Build a responsive image slider using jQuery

Make sure your images are displayed to maximum effect, regardless of the device the viewer is using

tools | tech | trends Dreamweaver, FlexSlider **expert** Neil Pearce

On the CD
- Start.zip
- Finish.zip

No matter what your proficiency, creating responsive designs can be a little confusing because of the radical change in thinking. As time goes on, responsive web design is drifting away from the pool of passing fads and rapidly entering the realm of standard practice. Simply put, this is a very different way of designing websites that represents the future. The web has moved beyond the desktop, and isn't looking back.

Mobile traffic has exploded over the past few years, and the number of devices we're designing for is growing just as fast! The changing landscape of web browsers matches that of users' expectations; people expect to be able to browse the web on the move just as easily as they do on a desktop. So in this tutorial we will add one useful tool to our belt – a responsive image slider – using the open-source jQuery plug-in FlexSlider.

01 Getting started

To begin, locate and unzip the 'start.zip' folder that is supplied on the CD. Here you will see an 'index. html' file which is a basic HTML5 template for us to start from. We also have empty JS and CSS folders and an image folder that contains just the 'bg.png' image file. Open up the index.html file in your text editor of choice and take a little look at the markup.

```
001  <!DOCTYPE html>
002  <html>
003  <head>
004    <meta content="charset=utf-8">
```

02 Viewport meta tag

Inside the head of our document we have a meta tag called viewport. The viewport tag is there to tell the browser that this webpage is optimised for mobile devices. By setting the viewport width equal to device-width, we are essentially telling it that the device width is 320 pixels. The initial-scale property controls the standard zoom level, while the maximum-scale and user-scalable properties dictate the degree to which users can zoom in and out of the page.

```
001  <meta name="viewport"
content="width=device-width; initial-scale=1.0;
maximum-scale=1.0; user-scalable=0;">
```

03 Head links

Underneath the viewport tag we have linked up to the jQuery CDN that we will need later, as well as a stylesheet that we will be creating in a few steps' time.

The only other thing to say here is that you will notice we have made some comments above our links. It's not required but is certainly encouraged and we will be adding comments throughout.

```
001 <!-- jQuery -->
002 <script src= https://ajax.googleapis.com/
ajax/libs/jquery/1.6.2/jquery.min.js></script>
003
004 <!-- main styles -->
005 <link rel="stylesheet" href="css/styles.
css" type="text/css" media="screen" />
```

04 Section tag
And last of all we have used the HTML5 section tag for our main wrapper. Even though HTML5 isn't our main focus here, it's never a bad thing including it wherever you can. Again, make sure the closing section tag is commented so if the page gets messy, we can easily find where the wrapper ends.

```
001 <section id="wrapper">
002 </section><!-- .END Wrapper -->
```

05 Default CSS
Create a new CSS file called 'styles.css' and save it into the CSS folder. Open the styles.css file into your text editor and add in our reset. The reset also covers some generic rules that are generally self-explanatory. However, the only rule that needs to be pointed out is the HTML5 display-role that is used to enable older web browsers to recognise any HTML5 elements as block level, like our section tag.

```
001 /* reset */
002 * {
003   margin: 0;
004   padding: 0;
005 }
```

06 Wrapper and body
For our wrapper rule, we need to make it 80% of the browser's viewport and give it a maximum width of 800px. Doing this will allow it to be scalable from 800 pixels down. We then centre it using margin: 0 auto. Straight underneath the wrapper rule we can add in our background image within the body tag using the 'bg.png', which as mentioned earlier is already located within the image folder on the disc.

```
001 #wrapper {  width: 80%; max-width:800px;
margin: 0 auto;}
002
```

```
003 body {
004
005   background: url('../images/bg.png');
006   font-family: Verdana, 'Arial', sans-
serif;
007 }
```

07 Find a font
Now jump back into the index.html file and, underneath the wrapper section tag, create an opening and closing h2 tag and type in a title. Now let's head over to www.google.com/webfonts and do a search for a font called Lobster. Add this to your collection and copy and paste the link within your document's head underneath the <!-- Google fonts --> comment.

```
001 <!-- Google fonts -->
002 <link href='http://fonts.googleapis.
com/css?family=Lobster' rel='stylesheet'
type='text/css'>
```

08 Finish the logo
Open up the styles.css file and copy and paste the font-family from Google Web Fonts. Then give it a font-size of 2.3ems and a small amount of top margin (we've used 50px). Once that's done, give the title a similar colour to our background. Finally, we apply a white text-shadow to give the text a bit more depth.

```
001 h2 {
002   font-family: 'Lobster', cursive;
003   font-size: 2.3em;
004   margin: 50px 0;
005   color: #d1c8ba;
006   text-shadow: #fff 1px 1px;
007 }
```

09 Pick up FlexSlider
Now we need to download the FlexSlider. So head over to flex.madebymufffin.com and grab the latest version, which at the time of writing is version 1.8. Once downloaded, locate the zipped file, unzip it and open up the folder. You will see two JS files and a CSS file. You need to put them into their respective folders within your own directory, then link them up inside the head of the index.html file.

```
001 <!-- FlexSlider pieces -->
002 <link rel="stylesheet" href="css/
flexslider.css" type="text/css" media="screen"
/>
003 <script src="js/jquery.flexslider-min.
js"></script>
```


<Above>
• How our logo looks after adding in the Lobster typeface from Google Web Fonts

10 FlexSlider markup
Adding in the markup for FlexSlider is really very simple. First, start with a containing div (<div id="slider">) that we can use to position the whole slider if we need to. Then we just need to add a single containing element: div class="flexslider"; this is what we are going to use as our main hook for the JS code.

```
001 <div id="slider">
002     <div class="flexslider">
003
004
005     </div>
006 </div>
```

11 Slides list
The next step is to add an unordered list (<ul>) in which to place the images we are going to populate the slider with. Give the list a class name of 'slides' and add in three list items, for now; this can be expanded on later. If you go back and take a look within the styles.css file, you will remember that we used a reset to get rid of the default bullet points of the <li> item.

```
001 <ul class="slides">
002   <li></li>
003
004   <li></li>
005
006   <li></li>
007 </ul>
```

12 Adding the images
Incorporating our images is very straightforward and nothing new to us. The pictures we used here are sourced from www.sxc.hu, and are available at no charge once you take a short amount of time to open a free account. Underneath the top image we have added a caption and given it a class of flex-caption. The flex-caption class is controlled with the flexslider.css file, so you will need to open that up to make any changes that are necessary.

> **Responsive web design is a very different way of designing websites that represents the future**

Build a responsive image slider using jQuery

```
001   <ul class="slides">
002    <li>
003    <img src="images/vase_img.jpg" />
004   <p class="flex-caption">Captions and
cupcakes. Winning combination.</p>
005    </li>
006    <li>
007    <img src="images/flower_img.jpg" />
008    </li>
009    <li>
010    <img src="images/field_img.jpg" />
011         </li>
012   </ul>
```

13 Flexible images

One thing we should point out is that we don't need to specify a width and height to the images within the main markup. This is very important because they need to be scalable and are controlled within the flexslider.css file. So if you open up this CSS file, you will notice on line 21 a rule that gives each image a max-width of 100%, allowing this amount of flexibility.

```
001  21 .flexslider .slides img {max-width:
100%; display: block;}
```

14 JavaScript time

Now that we have our markup, CSS and images set, let's add the functionality of the slider so we can see it in action. So within the head of our document (just above the closing </head> tag) type in this JavaScript code. Once you have done this, click and drag the browser window up and down to see the slider become responsive to the different browser sizes.

```
001   <!-- Hook to the FlexSlider -->
002   <script type="text/javascript">
003    $(window).load(function() {
004    $('.flexslider').flexslider();
005     });
006   </script>
```

15 Slider options

Like all good jQuery plug-ins, you are given plenty of extra options to customise things to your liking. All the options are located on the FlexSlider website, but just for fun let's play around with a few. First let's change the default animation from fade to slide.

```
001   <script type="text/javascript">
002    $(window).load(function() {
```

```
003
004    $('.flexslider').flexslider({
005
006     animation: "slide"
007
008     });
009
010    });
011   </script>
```

16 A vertical twist

How about we add another option? Instead of the default horizontal slide animation, let's make it a vertical slide animation. Even though it's not the norm, it certainly adds some originality, and if you dig a little deeper into the advanced options, you will find you could have two sliders working together – one side going vertically and the other sliding horizontally.

```
001   <script type="text/javascript">
002    $(window).load(function() {
003    $('.flexslider').flexslider({
004
005     animation: "slide",
006     slideDirection: "vertical",
007
```
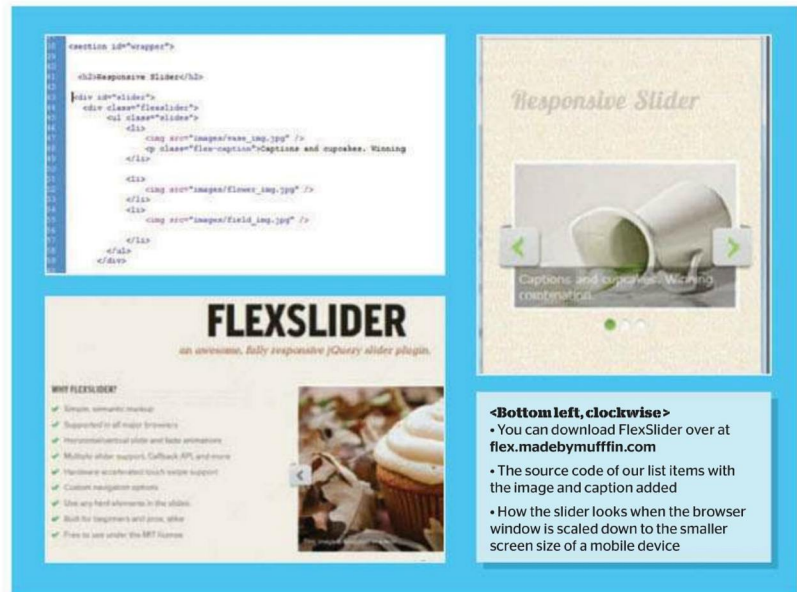
```
008     });
009
010    });
011   </script>
```

17 FlexSlider styles

It's always a good idea to open up the CSS when using any jQuery plug-in, so you can see what is doing what and perhaps customise the slider further. In this case we don't really need to do anything drastic. But one CSS rule worth pointing out is on line 35. These are the default styles of the background and they can easily be modified if needed.

```
001   /* FlexSlider Default Theme
002   ******************************/
003   .flexslider {
004    background: #fff;
005    border: 4px solid #fff;
006    position: relative;
007   -webkit-border-radius: 5px;
008   -moz-border-radius: 5px;
009   -o-border-radius: 5px;
010   border-radius: 5px; zoom: 1;
011
012   }
```
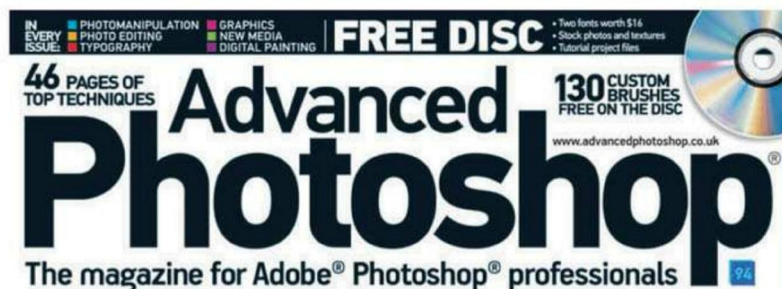
18 Final thoughts

Image sliders play a major role in today's web industry, and should be a feature of any designer's repertoire. When you have an existing website with one that needs to be made responsive, having the FlexSlider plug-in at your disposal can save you an awful lot of hard work – like any good plug-in should.

<Bottom left, clockwise>
• You can download FlexSlider over at **flex.madebymufffin.com**

• The source code of our list items with the image and caption added

• How the slider looks when the browser window is scaled down to the smaller screen size of a mobile device

> ❝ Like all good jQuery plug-ins, you are given plenty of extra options to customise things to your liking ❞

web workshop

Masonry layouts with How It Works

inspiration **www.howitworksdaily.com**

The masonry approach to design has become popular over the last few years because of its modular nature that allows the layout to reflow to devices of differing widths.

Using the Masonry plug-in differs from just basic reflowing of content though, as it allows the layout to fit together so that each part of the design is filled with the best fit for the content; just try resizing How It Works' homepage to see it in action. Obviously the result for each device is a layout that fits to the screen perfectly and so the user doesn't even realise that the layout is customised solely for their screen size. Blissfully unaware of the hard-working backend, they're presented with a full-screen, well-designed page every time.

Fitting the content together

The **How It Works** site has a variety of different sized images and differing amounts of text. The flexibility of the masonry style layout means that there aren't chunks of wasted space beneath each of the content blocks. Instead the content fits together in such a way that means there is less scrolling for the user.

1 INSPIRATION

Modular design

As our websites are now viewed on a plethora of devices, from the traditional desktop view to a range of smartphones and tablets, it has become increasingly difficult to know exactly how the site will be interpreted. Catering for differing dimensions can be a headache for many designers who are using techniques like CSS media queries and responsive design methods. The homepage of the **How It Works** site adopts a modular approach that allows segments to fill the width of the browser, forcing the remainder further down the page. This is a great solution for displaying on mobile devices.


Fitting to screens

Resizing the browser causes the masonry layout to reflow to fit the width of the browser, essentially solving the problem of differing sized screens.

Advert positioning

Adverts are positioned within the structure by making them the same width and height as the posts; this integrates them much more closely with the editorial.

Resize and reshape

When the browser is resized due to different devices, the modules animate nicely into their new positions via jQuery using options in the Masonry plug-in.
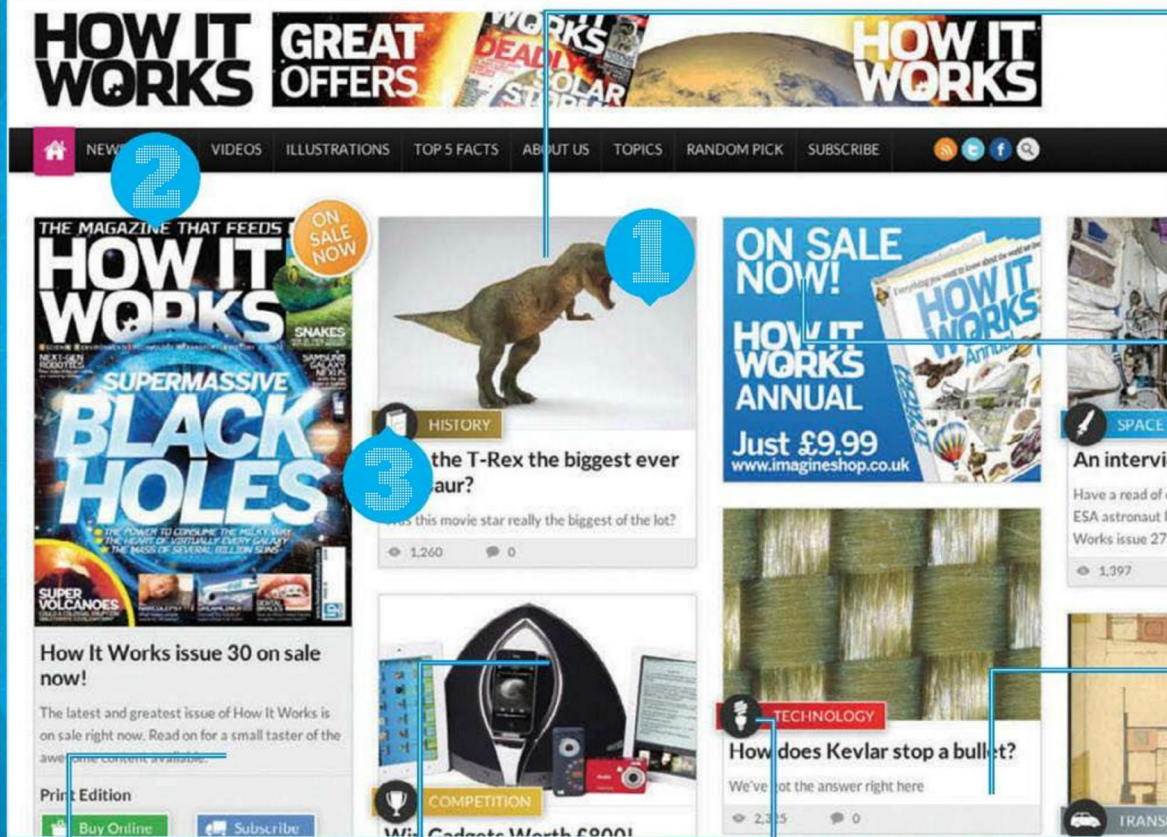
Length modules

Each post has a length module that varies in size. These modules slot together to create the core basis of the masonry layout.

WordPress

Each module that appears on the homepage is a post from a WordPress-powered blog. The design is customised to fit this kind of page setup.

Categories

Like with most blogs, posts are defined in categories. These use unique colour and icons to code each section and also tie in with the print version of the magazine.

2 TECHNIQUE Develop a masonry layout

01 Get the code

Download the latest version of jQuery, and get the Masonry plug-in from **masonry.desandro.com**. Create a new HTML document and save it into the same folder as your jQuery and the Masonry plug-in. In the head section add the following CSS, which creates a container to add each masonry element as a box.

```
001 <style>
002 #container {
003   padding: 5px;
004   margin-bottom: 20px;
005   clear: both;
006 }
007
008 .box {
009   margin: 5px;
010   padding: 5px;
011   background: #D8D5D2;
012   font-size: 1em;
013   line-height: 1.4em;
014   float: left;
015   -webkit-border-radius: 5px;
016   -moz-border-radius: 5px;
017   border-radius: 5px;
018   font-family:Arial, Helvetica,
sans-serif;
019 }
```

02 Finish the CSS

The following CSS code displays the images. Each column size is set up at 20px less than a multiple of 100. The clearfix code enables each of the boxes to fit together neatly in the masonry style.

```
001 .box img {
002   display: block;
003   width: 100%;
004 }
005 .col1 { width: 180px; }
006 .col2 { width: 280px; }
007 .col3 { width: 380px; }
008 .clearfix:before, .clearfix:after {
content: ""; display: table; }
009 .clearfix:after { clear: both; }
010 .clearfix { zoom: 1; }
011 </style>
```

03 Link to the code

Now we add links to both the jQuery and the Masonry plug-in. Make sure that both of these are saved in the same folder as your HTML. You might also want to put an image folder named 'img' with some sample images in. As we're organising the directory structure just now, this is a good time to add it.

```
001 <script src="jquery-1.7.1.min.js"></
script>
```

```
002 <script src="jquery.masonry.min.
js"></script>
```

04 Create some body content

Add this code to the body section of the page. The outer container holds all the masonry boxes. Notice here we have two boxes, but you can add as many as you like with the class of box col1, box col2 or box col3. The number represents column width.

```
001 <div id="container"
class="clearfix">
002   <div class="box col2">
003     <img src="img/sunflower.jpg"
alt="sunflower" />
004   </div>
005
006   <div class="box col1">
007     <p>Put some sample text in here.
</p>
008   </div>
009 </div>
```

05 Final formatting

Finally, we place the JavaScript shown below inside the closing body tag. This checks the images are loaded and ensures each is given the box selector. Set the column width to 100px here so columns will increase in multiples of 100. Save and test in your browser to make sure that all the boxes fit together.

```
001 <style>
002 var $container = $('#container');
003 $container.imagesLoaded( function()
{
004   $container.masonry({
005     itemSelector : '.box',
006   columnWidth: 100
007   });
008 });
009 </script>
```

Column sizing
Masonry-style layouts work equally well with different sized columns, as shown here, or you can keep with the fixed column style as demonstrated on the **How It Works** website.

3 TECHNIQUE

Designing custom icons

One of the nice touches to the design of this site is the icons that appear next to each entry in the layout. These build up a sense of recognition and categorisation that helps the user to navigate the content.

01 Get the typeface

Creating icons can take a huge amount of time in Illustrator, so you could just use a typeface that will kick-start the process! A great one to try is Heydings Icons which can be downloaded from **www. heydonworks.com/a-free-icon-web-font**.
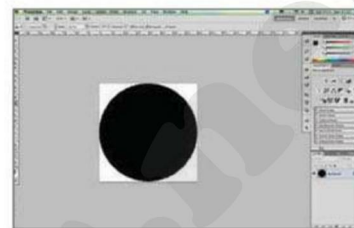
02 Into the image editor

The typeface downloads as a True TypeFace, so should be readable by most computers. Install it and open an image editor such as Photoshop. Create a new document and add a black circle using the Ellipse tool and fill with the Paint Bucket.

03 Apply your icon

To add the icon, select the Type tool you want to use and select white as the type colour. Then select the typeface 'Heydings icons' and type in a character. Scale the typeface up as necessary to fit the circle. Delete the background layer and save the icon as a transparent PNG file.

web workshop

Digital scrapbook styles in-site

inspiration **www.arminmorbach.com**

We all appreciate how styles that replicate real-world effects always go down well with today's web users. This is often seen in commercial sites, where the client wants a brand to reach out and connect with visitors. But beyond that, such styles are also applied by creatives in hope of enhancing a portfolio or page space, making it artistically more in-tune.

Adding illustrative elements is a certain way to achieve this, and mixed-media techniques can add authority if done in a sophisticated manner. Found images and according effects, such as torn paper textures and freehand font styles, give a sense of creative expression. The site of Armin Morbach (**www.arminmorbach.com**) is a fine example.

Intimacy with his site, however, is not just created with visual styles, but also with audio and real-time interaction (eg the floating interface), reflecting his brand's sensibility. This takes this web presence into another dimension – into the realms of multimedia.

Throughout this workshop we'll focus on delivering the graphic techniques and effects embraced by this website, enabling you to re-create similar scrapbook styles.

3D effects in 2D styles
A 3D collage effect may seem contradictory by nature, but it does make the most of contemporary UI animation. Below are some other effects that also work in this format...

1 TECHNIQUE

Drop shadow effects

01 Another dimension
Applying Layer Style>Drop Shadow is a subtle effect, but it will stop your design from looking flat and two-dimensional. Always apply from the second tier of Blend Mode options – ie Multiply, Color Burn or Darken. Set the Size so your shadow is tight to the model, and set the Spread at a value that maintains a visible edge.

02 Preset lighting
The Drop Shadow option is a great way to add unified lighting. Saving your settings as a New Style stores it as a preset, making it re-applicable to other elements in the Layer Styles>Styles options. If you want to manually tweak the effects go to Layer>Layer Style> Create Layer, which separates your shadow into its very own layer.
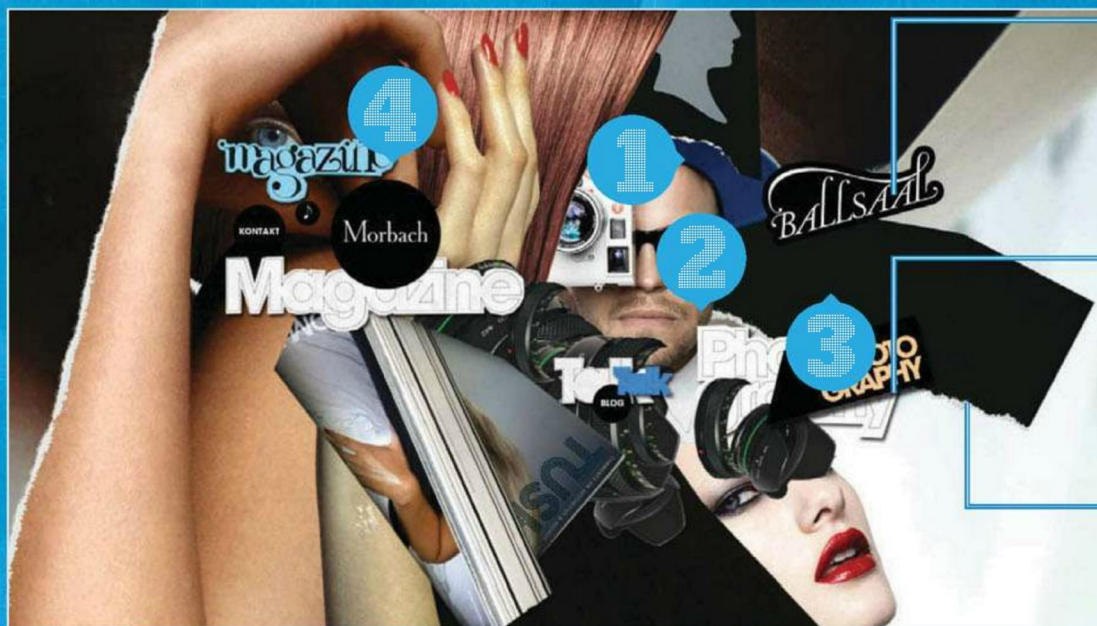
Blasé font styles
A mixture of fonts and typography enhance that scrapbook look, having an almost blasé attitude towards design. However, style cohesion is assured through juxtaposing font weights, sizes and colours.

Depth perception
Varying depths creates visual interest, constructed through layered floating elements. Combined with drop shadows some of the content is hidden to add some mystery to the site, as well as a collage feel.

Torn edges
The torn edge effect is what adds most authenticity to the overall style. The collage method is not just creative but also ties in with the expressive nature of both type and layout.

2 TECHNIQUE

Create digital rips with ease

There's nothing like an authentic look to scream 'real-world' in your digital projects. Tangible elements are always a big bonus in a creative arena that is seeing a backlash to synthetic styles. A torn edge effect can produce that tangibility, at the same time letting you retain your glossy photo images and clean graphics. It is easy to replicate such a style using Photoshop, with the help of some selection, brush and layering tools.

01 Cut out the model
Start by separating your model from the backdrop; this can be done as roughly or smoothly as you wish. For a more perfect edge use a Pen Path selection, otherwise use Lasso tools for something more immediate.

02 Brush styles
Open the Brush Preset menu, select the Splatter 14px style from the Brush Tip shape menu, then select Shape Dynamics. Set Size Jitter Control to Pen Pressure, and all Jitter controls to 100% with Minimum Roundness at 1%.
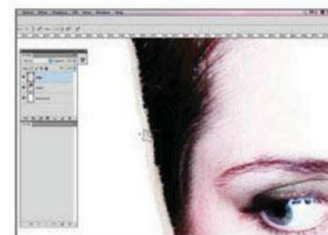
03 Paint your effect
Cmd/Ctrl-click your model layer thumbnail, create a new layer, then paint in your effect with a light grey brush. Invert (Cmd/Ctrl+Shift+I) and paint to the outer edge of the subject. Use both dark and light shades for a mottled effect.

3 TECHNIQUE

Dotty typographic work
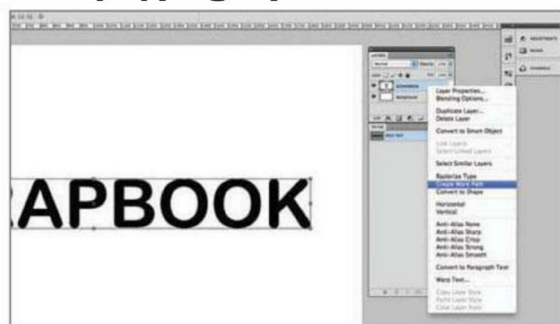
01 Convert the type
Begin by selecting the font you wish to use and spell out your word. We've used Arial Rounded MT Bold for its smooth edge. Next, Ctrl/right-click your layer and select Create Work Path, generating a Work Path in your Paths palette. Next, delete the type layer and create a new blank layer, leaving only your Work Path.
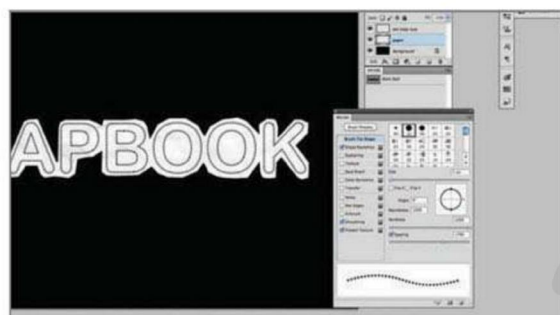
02 Optimise brush settings
With a hard brush, set the Size and Spacing to create sequential dots. Opt/Alt-click the Work Path, then Ctrl/right-click it, selecting the Stroke option. Lastly overlay your dot type layer on imported paper textures, and work up rough edges using the Pen or Lasso tool.

Long-term solution
Some web designers may consider a scanner surplus, and perhaps costly, but there's nothing quite like controlling your own content library. In the long run it may even work out cheaper than the constant purchase of premium online stock.

4 RESOURCE

Scanners and stock

If you do want to get away from artificial effects and actually use real-life clippings and collages, then a suitable scanner has to be a serious consideration. Specifically, photo scanner types are what you should look at – preferably high-resolution for those detailed digital projects.

The Canon CanoScan 9000F is a solid option. It boasts a whopping 9,600 x 9,600dpi optical resolution that easily trumps most competitors. This device also supplies users with the ability to scan photo film – 12 x 35mm filmstrip and 4x mounted slides, both positive and negative – while the Auto Photo Fix II and Auto Document Fix software optimises the output and quality of final images. Available from around £149 it won't break the bank either.

But if you don't want to bother with such an investment, but still want authentic stock then you couldn't do any worse than running a search on **naldzgraphics.net**. In the Freebies section, you can find a mountain of resources. Here the Naldz Graphics team have compiled over 200 high-quality paper textures, including ripped and distressed types, for anyone to grab. However, note that in commercial projects permission may still be required.

Tech tip
Bézier Curves
Piecing together imagery in collage styles is a given, but if you want to get the best precision selection then Photoshop's Pen tool is the best weapon. Other selection tools such as Marquee and Lasso don't even come close. The Pen tool easily masters edges with its Bézier Curves tech and editable anchor points. However, do practise drawing widely spaced anchor points, and experiment with shaping curves by adjusting the length and angles of the direction lines.

I ❤ Blog Beautiful
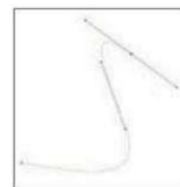
Creative agency blogs

Creative agencies are at the forefront of establishing some of the best-looking web

The blog categories stray from the typically right-sided mindset found on many blogs and can be found as a drop-down menu in the top-left.

Big bold backdrops add colour to this blog's page. These are randomly generated ensuring the user gets a different background upon reloading.

Work at Play

www.workatplay.com/blog

Development platform Drupal

On the face of it the Work at Play blog is very much a standard issue, albeit well-designed, site. However, on closer inspection the format is far more interesting. The blog layout is aligned to the right and the background image picks up the slack created on the left. The menu system positioned to the right-hand side of the screen is fixed for easy continual access.

Pod1

http://blog.pod1.com
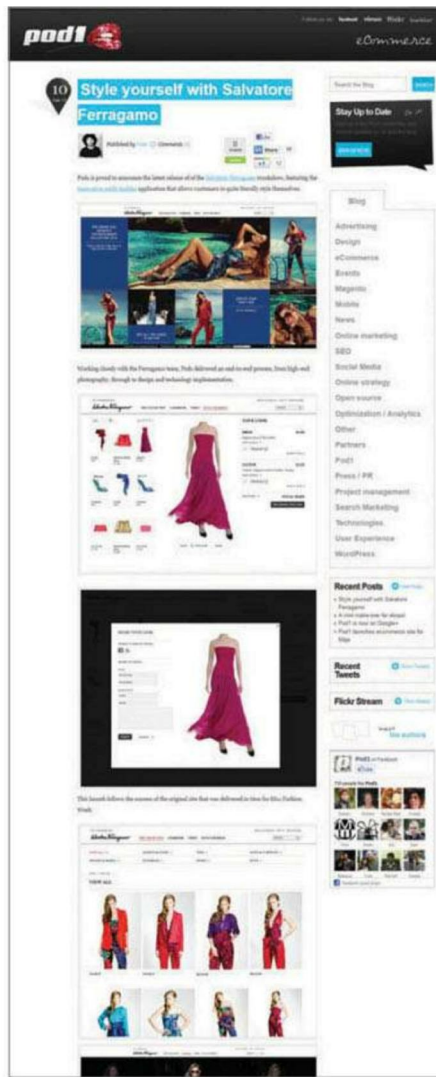
Development platform WordPress

All the principles of a blog are observed here, from the two-column layout to a right sidebar, but the styling and content make sure that it's a great example of the art. The background colour on the title and the dashed borders are neat touches. But what gives this blog its edge is the consistent use of great imagery, elevating it above the majority.

Internet Dreams

www.netdreams.co.uk/index.php/blog

Development platform WordPress

A dark and broody blog is given an instant lift with a collection of big, bright backgrounds that take the whole thing to another level. The blog itself is the standard two-column layout (including the sidebar), but a decorative header again lifts the gloom. To complete the ascent to blog stardom, a collection of bright, contrasting post images really make their presence felt.

experiences. Here we choose six blogs that showcase their undoubted talent

The featured images in each of the individual posts automatically zoom in the moment that the cursor is placed over the post box.

By default all the blogs are displayed. Filter Posts provides a drop-down list of specific post types for viewing a favoured category.

Neatly tucked in the top-left corner of every post is a link to multimedia content, containing the short form of its origin city.

❝ The Ultranoir blog steps outside the standard layout and stirs in a little imagination to create a visual feast ❞

Ultranoir
www.ultranoir.com/en/#!/blog/home
Development platform HTML, CSS, JavaScript

The Ultranoir blog is simply stunning. It steps outside the standard layout and stirs in a little imagination to create a visual feast. A grid keeps the assorted sized blog posts tight. A rollover adds a 3D effect to each post while the dynamic images add even more interest. The stacking of the fonts rounds off what is truly a thing of beauty.

45royale
www.45royale.com/blog
Development platform WordPress

A bright and engaging header image sets the tone for this blog. Adopting a three-column layout, a collection of Sans Serif fonts, along with a selection of appealing images and the perfect amount of white space makes this a shining example of what a blog can - and should - be. Navigation is conveniently and neatly tucked away in a drop-down list blending with the overall design.

Wieden+Kennedy
www.wk.com
Development platform HTML, CSS, JavaScript

The design of the WK blog is verging on minimalist with white space making its presence felt. The layout is sharp, well thought out and includes simple but effective design elements like a rollover effect which reveals titles and text. The blog is left aligned and complemented by a selection of right-sided backgrounds, making it resolution-friendly.

web {developer;

Dedicated to the code-heavy side of modern online design

Build mobile Android apps with PhoneGap

Part 1 of a practical look at forging native Android apps with this HTML5-based platform. **Page 82**

NEW 15-page section

Multi-threaded coding with the Web Workers API

The first part of our starter guide to running multiple processes. **Page 86**

POSTMODERN MODERNIZR 2.5

Discover updated browser detection features and how they fit into the new era of responsive design. **Page 76**

POSTMODERN MODERNIZR 2.5

BROWSER DETECTION REDEFINED

Louis Simoneau looks at how this newly updated open-source JS library adapts to a fresh era of super-flexible web design

he term 'responsive web design' as it's most commonly used refers to designing and building sites that adapt to the user's screen size. But any way in which your site responds to the conditions it finds itself in (the user's browser, in other words) could equally be termed responsive design.

In this article, we'll be looking at how your site can adapt not only to the browser's window size, but also to its capabilities – an important concern in a browser market that's seeing new, experimental features introduced at an unprecedented rate. What kinds of situations might this approach be applied to? Some examples are:

- A site using box-shadow to offset an aside or a boxout might want to add a border to the element if box-shadow is unsupported
- You might want to use an image and CSS image replacement for your site's title if the browser doesn't support @font-face
- You may use a JavaScript library to 'fake' support for an HTML5 element, such as <meter>, or a feature such as Web Storage or Canvas. However, the library should only be loaded when the browser lacks support for the corresponding feature, so as not to penalise users of more recent browsers with a superfluous download
- As part of a responsive layout, you might want to load a JS-based 'feature slider' only when the browser's width is above a certain threshold.

As you can see, the ways in which your site can be made to respond to the browser's capabilities in addition to its size are numerous, and vary from those that are extremely simple to others which are far

more complex. Fortunately, they're all made possible by a single powerful tool: Modernizr.

WHAT IS MODERNIZR?

Modernizr was originally written by Faruk Ateş in 2009, and at the time of writing is in version 2.5.3.

It's a small bundle of JavaScript that provides simple feature detection for pretty much every new browser technology, from HTML5 to CSS3 to JavaScript APIs like Drag and Drop, or Local Storage.

Modernizr operates in two distinct ways: first, by adding classes to the <html> element, and second by providing a JavaScript API.

MODERNIZR FOR CSS

When you first load a page that includes Modernizr, the library performs its set of feature detections, and uses the results of this to add class attributes to the page's <html> element.

For example, here we can see a list of the classes added in Firefox 10:

```
001   <html class="js no-flexbox canvas
canvastext postmessage no-websqldatabase
indexeddb hashchange history draganddrop
websockets rgba hsla multiplebgs backgroundsize
borderimage borderradius boxshadow textshadow
opacity cssanimations csscolumns cssgradients
no-cssreflections csstransforms csstransforms3d
csstransitions fontface generatedcontent video
audio localstorage sessionstorage webworkers
applicationcache">
```

Every property that can be detected in Modernizr is included, but those properties that are unsupported in the browser you are using are prefixed with 'no-'. So, say we are using Firefox 10, we can see that a large number of CSS3 and HTML5 features are supported, and only a few - like FlexBox, CSS reflections and WebSQL – are not.

The idea is to use these classes in your CSS to provide different styles to browsers depending on their level of support for targeted features.

For example, the current version of Firefox supports CSS columns, but many older browsers don't. Here's how you'd use the Modernizr-provided classes to make your pages responsive to the presence or absence of column support:

```
001  p {
002    /* general-purpose paragraph styles */
003  }
004
005  .no-csscolumns p {
006    /* styles for paragraphs when CSS column
support is absent */
007  }
008
009  .csscolumns p {
010    /* paragraph styles for use with CSS
columns */
011  }
```

Because Modernizr adds classes both when a feature is present and when it's absent, you're able to provide styles for both cases, as well as default styles that will apply in both cases.

MODERNIZR FOR JAVASCRIPT

The CSS classes are Modernizr's most visible feature, but they're only a small part of what it makes possible.

In addition to the classes, Modernizr provides you with a JavaScript interface for querying which features are available. It all happens through the Modernizr object, which is made available simply by including Modernizr in your page.

The Modernizr object has a property for each feature that it detects, which will simply be either true or false depending on whether or not the feature is supported. There are far more properties that can be accessed via the JavaScript API than the number of

DETECTING FEATURES VS BROWSER SNIFFING

It's important to note that when you use Modernizr, you're detecting for the presence of specific features in the browser, not which browser you're dealing with. If you've read much about web design, chances are you'll have been sternly warned about 'browser sniffing', and with good reason: browsers can lie about their identity, and even the same browser can behave differently on a different OS or with a different configuration. Browser sniffing is also not future-proof. Just because version 9 of a browser has a certain feature doesn't mean that every future iteration will as well. So tying your site's adaptive features to a specific browser version is, in many ways, akin to a ticking time bomb.

By detecting features instead, you guarantee that you're adapting your website to the real context that it's being used in, while at the same time ensuring that the site behaves as intended in any future browsers that it may encounter.

classes added for CSS purposes (and, just to give you an idea, more than 60 new detects were added in version 2.5 alone), so it's best to consult the documentation at **www.modernizr.com/docs** to find out about all of them.

Use the properties in conditional logic just as you would any other Boolean (true or false) value. For example, you could use Modernizr to determine if the user's browser includes support for the popular Geolocation API as follows:

```
001  if (Modernizr.geolocation) {
002    // code that makes use of geolocation
003  }
004  else {
005    // fallback, perhaps asking the user to
enter their location instead
006  }
```

Of particular interest in the context of responsive design, Modernizr's JavaScript API also provides tests for media query support, as well as allowing you to test specific queries, via the mq() method.

For example, the following test will return true for windows no more than 768px wide:

```
001  Modernizr.mq('only screen and (max-width:
768px)')
```

This can be a powerful way of adapting your JavaScript to pair up with responsive CSS.

MODERNIZR FOR LOADING RESOURCES

So far we've seen how Modernizr can make both CSS and JavaScript more responsive to either the browser's capabilities or simply its dimensions, but Modernizr has yet another trick up its sleeve.

A common technique for dealing with varied levels of browser support for a given feature is to use polyfills to patch up holes. A polyfill (a term coined by developer Remy Sharp) is a piece of code that makes it seem as though the browser natively supports a feature that it doesn't. That way, rather than writing your own complex JavaScript workarounds for missing functionality, you simply use the functionality assuming it's present, and include a polyfill to bring older browsers up to speed. There are polyfills available for everything from CSS3 styles to Web Sockets to SVG, and there's a growing list on the Modernizr wiki (**bit.ly/ddLS2d**).

However, the use of polyfills does come with one caveat, which, as we'll shortly see, Modernizr helps solve. The problem is that, when you include a JavaScript file in your pages, it will always be downloaded. That is, even browsers that support the feature you want to polyfill will spend time downloading an extra chunk of JavaScript they'll never use. We want our pages to be snappy and light, not bandwidth hogs, so this is an issue.

Modernizr helps solve this problem with the load() method. Load() lets you conditionally load resources (CSS or JavaScript) depending on the outcome of a feature test (or, in fact, depending on any condition you can express in JavaScript). Here's an example of Modernizr.load in action:

```
001  Modernizr.load({
002    test: Modernizr.websockets,
003    yep: 'websocket-chat.js',
004    nope: ['sockjs-0.2.min.js', 'sockjs-chat.
js']
005  })
```

In this example, assume we've got a little JavaScript chat client, written to use HTML5's Web Socket capabilities (websocket-chat.js). However, realising that not all browsers support Web Sockets, we've

MODERNIZR DOESN'T REPLACE GRACEFUL DEGRADATION

RGBA wasn't the only place that IE8 failed in rendering our note. The subtle box shadow on the note itself is also missing. While your instinct might be to use Modernizr to apply some workaround styles to provide the same appearance (say with a semi-transparent repeated background image), that's not always the right approach.

The traditional best-practice approach of graceful degradation is a much better strategy in this case: the note still looks fine without the shadow after all, so why spend hours crafting a workaround that will only ever be seen by a small subset of your visitors (a subset which will, moreover, shrink even further over time as those users upgrade to newer browsers)? The answer is, you shouldn't: degrade gracefully where you can, and only use Modernizr to get you out of trouble when there's no other option.

written another version of the client using SockJS. SockJS (**github.com/sockjs/sockjs-client**) is a library that attempts to replicate all of the functionality of Web Sockets in older browsers (or behind firewalls that prevent Web Sockets from functioning). The SockJS client is located in the file 'sockjs-0.2.min.js', and our version of the chat client using SockJS instead of stock Web Sockets is in 'sockjs-chat.js'.

Modernizr.load accepts as its parameter a JavaScript object literal, containing a few key-value pairs. There is a fair number of options you can use, but the most important for now are 'test', 'yep' and 'nope'. The first is the test to run to determine what to load, while yep defines what will be loaded if the test returns true, and nope defines what will be loaded if the test returns false. As we've done in the example above, yep and nope can either be a single filename or an array of multiple filenames.

Thanks to load(), our users on cutting-edge browsers won't be penalised with a useless download, and everyone still gets the same functionality.

A SIMPLE EXAMPLE

Now that you have got a basic understanding of Modernizr's various features, it's time to put it to the test. The best way to get your head around how Modernizr works is, very simply, to start using it. Let's look at a relatively basic example first, followed by a more involved one.

Our goal in the first example is to use CSS3 to create a sticky tape effect, giving the impression that an aside is a note that's been 'stuck' to the site.

To create the tape, we'll use a technique developed by Michael Angeles (**konigi.com/tools/css-sticky-notes**). The tape is actually generated content created using the :after CSS rule on the aside:

```
001  aside {
002    margin-left: 15px;
003    padding: 10px 20px;
004    width: 250px;
```

```
005      background: rgb(244, 243, 158);
006      float: right;
007      position: relative;
008    box-shadow: 0 2px 2px rgba(0,0,0,0.2);
009     }
010
011      aside:after {
012      display: block;
013      content: "";
014      position: absolute;
015      width: 110px;
016      height: 30px;
017      top: -21px;
018      left: 30%;
019      border: 1px solid #fff;
020      background: rgba(254, 254, 254, .6);
021      box-shadow: 0px 0 3px rgba(0,0,0,0.1);
022    }
```


Use the Modernizr build tool to select just the features you want, for the leanest and meanest Modernizr possible, totally bespoke to you

Using the rgba() method for colours allows the sticky tape to be partially transparent for authenticity.

In browsers that support all the features used (the :after pseudo-selector, rgba() colours, box-shadow, etc), such as Firefox or Chrome, the sticky note looks just how you'd expect it to. In Internet Explorer 8, however, the picture is far less pretty.

The problem here is that Internet Explorer 8 and below won't apply any styles whatsoever to HTML elements they're unfamiliar with, including HTML5's new semantic elements, such as <aside>.

Fortunately for us, Modernizr comes bundled by default with a little snippet of JavaScript initially developed by Remy Sharp and commonly referred to as the 'HTML5 shiv', which allows for styling <aside> and its ilk in older versions of IE.

DOWNLOAD MODERNIZR

You can download Modernizr from **www.modernizr. com**. The development version will be fine for this example. It's much too large to serve from your production sites, but it's uncompressed so it's easy to
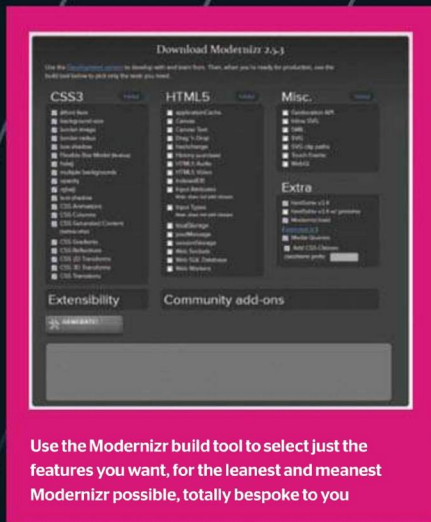
dive into the source when you want to find out what the library is doing behind the scenes.

While it's generally recommended for performance reasons to include JavaScript as far down the page as possible (near the closing <body> tag), Modernizr is an exception. This is because we want to rely on the classes it adds to the <html> element in our CSS, and because we want to rely on the HTML5 shiv to style some elements, it will need to come before any styles:

```
001   <head>
002      <meta charset="utf-8">
003      <title>Demo page for Modernizr 2.5</
title>
004
005      <script src="js/modernizr-2.5.3.js"></
script>
```

Already there's an improvement; the aside is correctly floated right, and its background is the yellow that we want. However, there are still a few problems. For instance, IE8 fails to understand the RGBA colour

declaration or the box-shadow, but it does understand the border rule, so the result is a white outline that looks decidedly broken.

This is precisely the sort of situation where Modernizr excels; when we don't want the browser to apply as many of the CSS rules as it can depending on its capabilities, we want everything or nothing because anything in between will look broken.

When you're using Modernizr, you always want to take a step back and think about which styles should be served under which conditions. So let's break down our little problem:

- Browsers that understand RGBA colour and box-shadow should display the tape
- Browsers that understand neither should omit the tape entirely
- Browsers that understand box-shadow but not RGBA colour shouldn't display the tape either. Even with a nicely feathered drop shadow, it will still be an empty outline
- Browsers that understand RGBA colour but not box-shadow should still show the tape. Even without the feathered edge, it still looks fine. This is a classic example of graceful degradation.

From this breakdown, it's clear that the deciding factor in this instance, in whether or not to display the tape, is support for RGBA.

To find out if Modernizr supports detecting RGBA, you can either flip through the list of detected features in the documentation (over at **www.modernizr.com/ docs/#s2**), or for greater convenience, use your browser's web inspector to look at the set of classes on the <html> element. Sure enough, when viewing the test page in Firefox, there's an RGBA class there, so we can expect to see the corresponding no-rgba class in Internet Explorer 8.

Using this class, we can ensure that non-compliant browsers don't make any attempt whatsoever to render the sticky tape:

```
001      .rgba aside:after {
002          display: block;
003          content: "";
004      …
005  }
```

A RESPONSIVE DESIGN EXAMPLE

Now that we've seen how Modernizr can be used to assist in our CSS wizardry, it's time to see what it can do to help us make our sites more responsive.

You'll surely have come across a 'featured image slider' of one description or another in your travels on the web. These are big, usually animated widgets that sit at the top of a page and cycle through a set of high-resolution images, often with accompanying captions. They're a great way to add visual punch to a page, and give it that 'magazine' feel.

But a set of 800px-wide images and the big wad of JavaScript required to make the slider run are the last thing you want mobile visitors to be downloading.

Fortunately, Modernizr's load() and mq() methods make it so that only browsers above a certain width will load the resources required for the slider.

```
001      <script src='https://ajax.googleapis.
com/ajax/libs/jquery/1.7.1/jquery.min.js'></
script>
002      <script>
003      $('document').ready(function(){
004          var widescreen = Modernizr.
mq('only screen and (min-width: 800px)');
005
006          Modernizr.load({
007              test: widescreen,
008              yep: ['nivo-slider/jquery.nivo.
slider.pack.js', 'nivo-slider/nivo-slider.css',
'nivo-slider/themes/default/default.css'],
009              complete: function() {
010                  if (widescreen) {
011                      $('article').before(
```

```
012                          ['<div class="slider-
wrapper theme-default">',
013                          ' <div id="slider"
class="nivoSlider">',
014                          '   <img src="images/img1.
jpg" alt="" />',
015                          '   <img src="images/img2.
jpg" alt="" />',
016                          '   <img src="images/img3.
jpg" alt="" />',
017                          '   <img src="images/img4.
jpg" alt="" />',
018                          ' </div>',
019                          '</div>'].join('\n')
020                      )
021                      $('#slider').nivoSlider();
022                  }
023              }
024          });
025      });
026  </script>
```

This example is using the popular Nivo Slider (available from **nivo.dev7studios.com**), an open-source jQuery plug-in for creating image sliders.

We start off by storing the result of the mq() test in a variable as we'll be using it more than once.

Then comes the Modernizr.load() call: we check the widescreen variable to see if the browser is over 800px wide. If yes, we load three resources: the plug-in itself, a base stylesheet and a 'theme' stylesheet. Notice there's no nope; that's fine – we don't want to load anything if the test is false.

Finally, there's a complete key. This is one we haven't seen before. It lets you define a function to run once all the resources have finished downloading (whether the test passed or not). In this case, we check widescreen again, and if it's true, insert the markup required for the slider, and call nivoSlider() on the slider div to make Nivo work its magic.

Because the img tags are inserted dynamically only when the browser window is above a certain

width, the images won't be downloaded at all in a smaller browser. Try it out; load the page, then resize your browser to below 800px and refresh the page. This time, there's no big fancy image slider, and the page load is nice and snappy.

CONCLUSION

Now that you've seen what Modernizr can do, it's time to put it to the test in your own projects. A few final words of advice before you do, though.

This should go without saying, but as Modernizr is a JavaScript library, those CSS classes won't be added if JavaScript is disabled.

Just because a feature can be faked with workarounds or polyfills doesn't mean it should. While you could use Modernizr to detect border-radius support and implement a complex, multiple-background-image workaround for rounded corners, why? Settle for square corners in a few browsers, and spend that time improving in more productive ways.

Now, get to it! With the power of Modernizr, you can craft websites that perform well on new browsers and old, small screens and huge: sites that always deliver the best possible experience to your users.
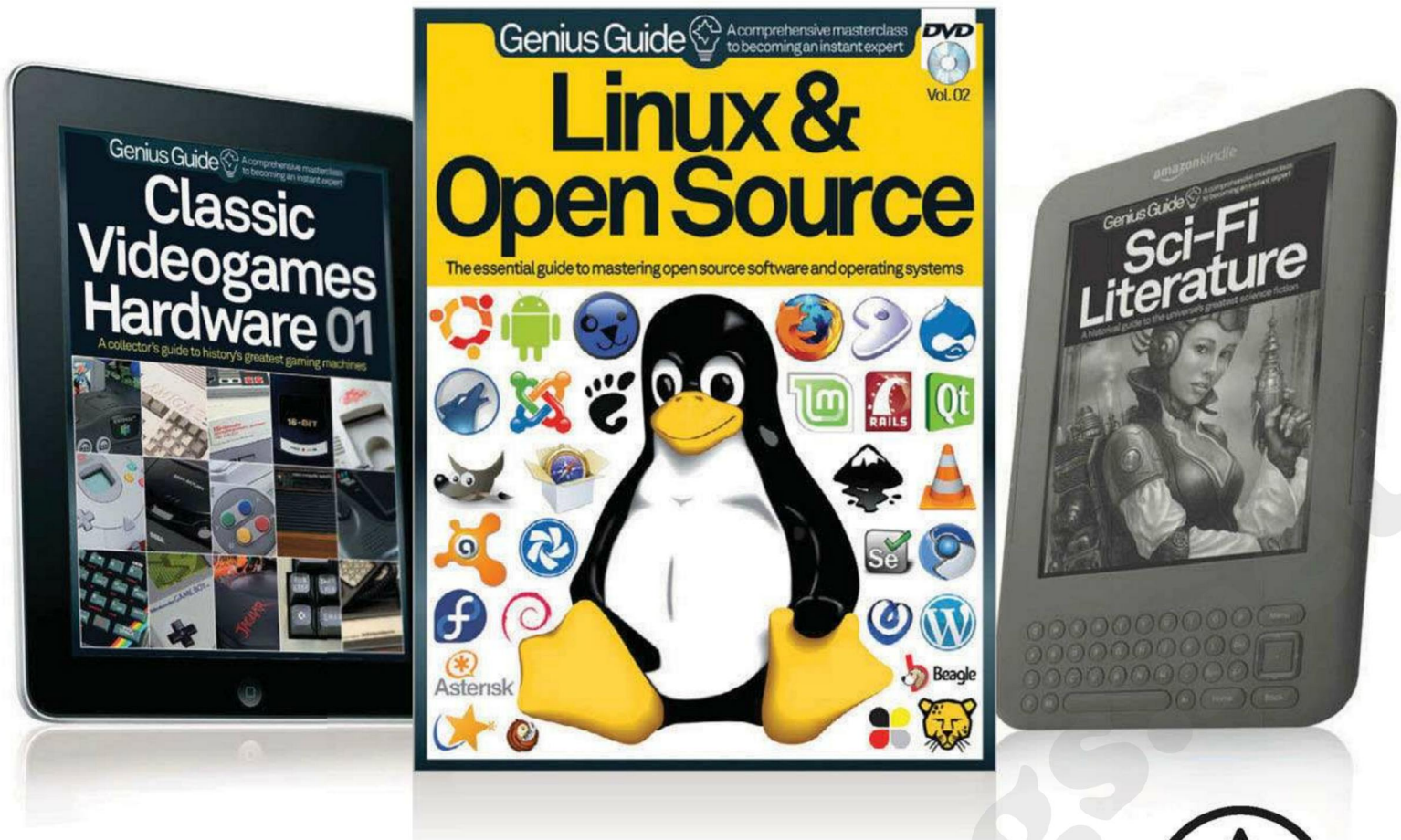
Build apps for Android with PhoneGap

Use your HTML, CSS and JavaScript skills to develop native mobile applications for Android devices

tools | tech | trends Eclipse, PhoneGap, ADT plug-in
expert Matt Gifford

On the CD
- Step code
- Completed project

PhoneGap is an open-source development toolkit that provides developers with the ability to easily interact with and manage a mobile device's native functions and processes, without having to go through the long and tiresome process of learning a new programming language. Instead, applications are built using HTML5, CSS3 and JavaScript, the latter of which integrates with the underlying API to manage the device.

This means that any web professional who has the ability to write simple JavaScript functions can easily create cross-device native apps that act and feel exactly like a traditional native mobile application.

In the first instalment of this two-part guide we take a look into setting up the right development environment for creating and debugging Android applications. We'll also create a simple application that utilises some of the API methods to detect device movement and to manipulate the DOM elements for data output.

---

01 Get the Eclipse IDE
For this tutorial we are going to use the open-source Eclipse project as our IDE. It's highly extensible with a wide array of plug-ins and extensions to enhance your development environment. If you have not already got a copy, download the latest build from **www.eclipse.org/eclipse**.

```
001  http://bit.ly/rlHTFK
```

02 Install Android plug-in
Once your Eclipse workbench is up and running let's download the Android Development Tools plug-in. Go to Help>Install New Software and click on Add to provide a new repository location. Set 'ADT' as the name, and the URL for the repository defined as per below. Select all available tools to install and follow the wizard to complete the process.

```
001  http:////dl-ssl.google.com/android/eclipse/
```

03 New project
Create a new project in Eclipse under File>New>Android Project; call it 'DailyQuote'. Under Build Target, select Android 2.2 and give it a unique application name – 'com.coldfumonkeh.quoteapp', for example. Create an activity called 'DailyQuoteActivity' and finish the project wizard. This will generate the required files and a structure similar to the following:

```
001  project structure:
002  src
003  gen
004  Android 2.2
005  Referenced Libraries
006  assets
007  bin
008  libs
009  res
010
```

04 Download PhoneGap
We now need to download the open-source PhoneGap API project. Visit **phonegap.com** to get the package. Extract the archive and navigate into the lib/android directory. Copy the entire XML directory from here and paste it into the /res directory within your Eclipse project. This defines the relevant libraries for each native device process.

```
001  <?xml version="1.0" encoding="utf-8"?>
002  <plugins>
003      <plugin name="App"          value="com.phonegap.App"/>
004      <plugin name="Geolocation"  value="com.phonegap.
GeoBroker"/>
005      <plugin name="Device"       value="com.phonegap.
Device"/>
```
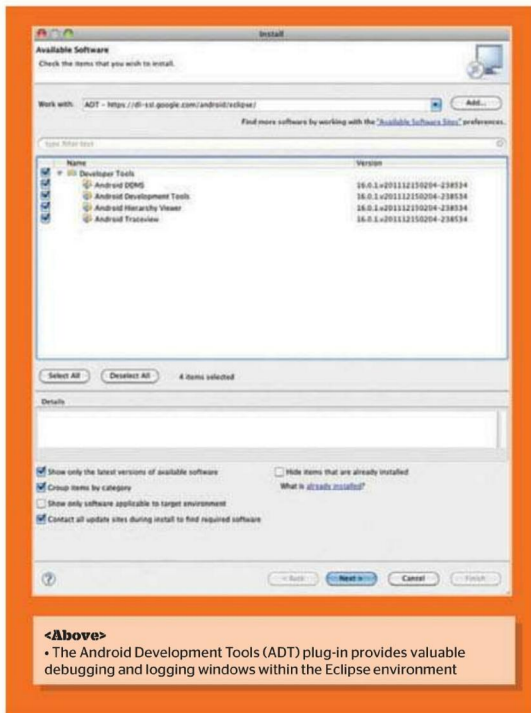
05 Add JAR library
Next, we need to import the PhoneGap JAR file into the Eclipse workspace. Copy the file and paste it into the /libs folder within our project. Now Cmd/right-click on the /libs folder, select Build Path>Configure Build Path and choose the Libraries tab. Hit Add JARs and locate the PhoneGap JAR file. Confirm all selections to complete this step.

06 Amend the Java package
The default Android Java package needs to be modified to draw from the PhoneGap component libraries. Open your activity *.java file within the project's SRC directory and change the package to extend DroidGap. We also need to import the PhoneGap components and load the index.html file to make it the default initial page.

```
001  package com.coldfumonkeh.quoteapp;
002
003  import com.phonegap.*;
004  import android.os.Bundle;
005
```

\<Above\>
• The Android Development Tools (ADT) plug-in provides valuable debugging and logging windows within the Eclipse environment

activity that references the PhoneGap DroidGap component library. Importantly, this activity also handles how the application reacts to changes in the device orientation. Include the new activity node within the application tag node block in the XML.

```
001 <activity android:name="com.phonegap.DroidGap"
android:label="@string/app_name" android:configChanges="orientatio
n|keyboardHidden"> <intent-filter> </intent-filter> </activity>
```

09 Create the HTML

Create a new file called 'index.html' in the /assets/www directory. This will act as the default view for our application. Write script references within the head tags to include both the jQuery and PhoneGap JS files.

```
001 <!DOCTYPE html>
002 <html>
003   <head>
004     <title>Daily Quote</title>
005       <link rel="stylesheet" href="style.css">
006       <script type="text/javascript" charset="utf-8"
src="jquery-1.6.4.min.js"></script>
007       <script type="text/javascript" charset="utf-8"
src="phonegap-1.4.1.js"></script>
008   </head>
009   <body>
010     super.loadUrl("file:///android_asset/www/index.html");
011
012 </body>
013 </html>
```

10 Define DOM elements

Now let's add the DOM elements within the body tag of the HTML file. We'll need three elements: a blockquote tag with the id attribute set to quote; a button with the id attribute set to refreshBtn; and a paragraph tag block with the rel attribute set to cite. We'll be using this trio of attributes to reference the elements using jQuery.

```
001 <blockquote id="quote" class="quoteBlock"></blockquote>
002 <p rel="cite"></p>
003 <button id="refreshBtn" class="super button pink">Show
Another</button>
```

11 Custom code

Below the existing JavaScript references, you will need to place a new script tag block, into which we'll write our custom code to interact with the PhoneGap API. Firstly, let's define some required global variables that we'll use later on in the coding process.

```
001 <script type="text/javascript" charset="utf-8">
002
003 var prevX = 1.0;
004 var arrQuotes = [];
005
006 </script>
```

12 Add an event listener

Within the empty tag block we can now define the first of our event listeners. The DeviceReady listener will fire once the PhoneGap code has fully loaded and is available to be executed. This will run a new method called onDeviceReady, into which we'll place the initial code for our application.

```
006 public class DailyQuoteActivity extends DroidGap {
007     /** Called when the activity is first created. */
008     @Override
009     public void onCreate(Bundle savedInstanceState) {
010     super.loadUrl("file:///android_asset/www/index.html");
011     }
012 }
```

07 Define AndroidManifest

Open up the 'AndroidManifest.xml' file from the project root. We need to define the permissions for our application to access and use the device functionality. This can include use of the device camera, internet connectivity and network state data, as well as write access to the device storage. Place this data below the opening manifest node.

```
001 <supports-screens
002     android:largeScreens="true"
003     android:normalScreens="true"
004     android:smallScreens="true"
005     android:resizeable="true"
006     android:anyDensity="true"
007     />
008     <uses-permission android:name="android.permission.CAMERA"
/>
009     <uses-permission android:name="android.permission.VIBRATE"
/>
```

08 Define AndroidManifest 2

Still within the 'AndroidManifest.xml' file, we also need to define a new

‹Above›
• For the application to interact fully with the PhoneGap API, apply the external JAR libraries into the project build path

```
001  document.addEventListener("deviceready", onDeviceReady,
     false);
002
003  function onDeviceReady() {
004
005        // more code to go here
006
007  }
```

13 Binding the button

Instead of assigning an onClick attribute to our button element to perform an action, we can utilise the power of jQuery and add a click handler to the button, which we can do within the onDeviceReady method. Here we are simply binding a click handler to the refreshBtn element, which will then run the refreshQuote method.

```
001  $('#refreshBtn').bind('click', function() {
002          refreshQuote();
003  });
```

14 AJAX call

Copy the 'quotes.json' file from the tutorial step code on the disc and paste it into the /assets/www directory within the Eclipse project. We are now able to add an AJAX request into the onDeviceReady method to read the file and store the quote array into the predefined arrQuotes variable. This means it will now run the getQuote method.

```
001  var request = new XMLHttpRequest();
002  request.open("GET", "file:///android_asset/www/quotes.json");
003  request.onreadystatechange = function() {
004      if (request.readyState == 4) {
005            var data = JSON.parse(request.responseText);
006            arrQuotes = data.quotes;
007            getQuote();
008      }
009  }
010  request.send();
```

15 Display the quote

To select a quote to display within our application, we'll obtain a random index from the array by determining the array length. This will pull out an individual quote object from the stored values within the arrQuotes variable. We can then display the quotation and the name of the person it's attributed to by referencing the DOM elements using jQuery.

```
001  function getQuote() {
002        var quoteObject =
003            arrQuotes[Math.floor(
004                    Math.random() * arrQuotes.length)];
005      $('p[rel=cite]').html(quoteObject.name);
006      $('#quote').html('<p>' + quoteObject.quote + '</p>');
007  }
```

16 Shake gesture detection

As well as allowing the user to refresh the quote using a click/touch gesture, we can harness PhoneGap's integration with the mobile device and make use of a built-in accelerometer to detect a shake gesture via a movement on the X axis. If detected, the refreshQuote method will also run.

```
001  function watchForShake(threshold) {
002      var axl = new Accelerometer();
003      axl.watchAcceleration(;
004      function (Accel) {
005      if (true === Accel.is_updating){
006            return;
007      }
008        var diffX = Math.abs(Accel.x) - prevX;
009      if (diffX >= threshold) {
010            refreshQuote();
011      }
012          prevX = Math.abs(Accel.x);
013      }
014      , function(){}
015      , {frequency : 750}
016    );
017  }
```

17 Shaken, not heard

With our accelerometer method written to detect the device movement, we need to include the reference to the method, as it is not being run from an event listener. Edit the onDeviceReady method and place the code to call the watchForShake method within the function.

```
001  watchForShake(0.5);
```

18 Refresh methods

We now have two separate calls to our refreshQuote method. The first is from the button and the second from the shake gesture detection. Let's create that function and simply insert a call within it to run the getQuote method which we have previously defined.

```
001  function refreshQuote() {
002      getQuote();
003  }
```

19 Graceful exit

Next, let's provide the user with a simple way to efficiently close down the application. Add a new event listener, this time looking for the click use of

the device's back button. Once logged, this will run a method to prompt the user with a yes/no dialog box to confirm or reject application exit.

```
001  document.addEventListener("backbutton",
showExitConfirm,  false);
002
003  function showExitConfirm() {
004      navigator.notification.confirm(
005          'Are you sure you want to exit?',
onExitConfirm,
006          'Why you no stay?',
007          'No,Yes'
008      );
009  }
```

20 Act on input

Regardless of the action selected by the user in the confirmation box, the results from the button will be passed through to the onExitConfirm function. From here we can determine their selection using an index number, and if the second button was chosen, we will swiftly close down the application.

```
001  function onExitConfirm(button) {
002      if(button == 2) {
003          navigator.app.exitApp();
004      }
005  }
```

21 Integrate browser

Let's provide users with the ability to further investigate quotes from specific individuals. We can set an event listener on the device's search button, which, if pressed, will use the navigator component and the loadURL method to open up a Google search page with the provided parameters that we send through.

```
001  html {
002      height: 100%;
003      background-color: #FFECD0;
004      background-image: -moz-linear-gradient(-45deg,
rgba(255, 255, 255, .3) 25%, transparent 25%, transparent
50%, rgba(255, 255, 255, .3) 50%, rgba(255, 255, 255, .3)
75%, transparent 75%, transparent);
005      + $('p[rel=cite]').html();
006      navigator.app.loadUrl(urlString);
007  }
```

22 Apply the style

To wrap up, create the stylesheet within the /assets/www directory in your Eclipse project and call the file 'style.css'. As we are dealing with standard HTML development, we can utilise CSS3 properties and create visuals in exactly the same way as we would when creating a typical web app. The full code is on the disc.

```
001      jsonScript.setAttribute("id", "jsonScript");
002      jsonScript.setAttribute("src", src);
003          head.appendChild(jsonScript);
004
005      setTimeout("insertScript();", 25000);
006  }
```


<Left to right>
• When running the application on the device, the output is clear, concise and styled thanks to a simple selection of CSS3 properties
• Clicking the Device Search button takes the user to a Google search result page for further info

Device Manager
We can debug or run the application build on a virtual Android device, created through the Device Manager tools, which we can set to act as a specific device with set commands and capabilities.

Code library
In detail
Let's take a closer look at some of the techniques used within the project to build our application

To determine changes in the acceleration of the device we call the watchAcceleration method to periodically check for movement every 750 milliseconds.

```
001  function watchForShake(threshold) {
002      var axl = new Accelerometer();
003      axl.watchAcceleration(
004      function (Accel) {
005          if (true === Accel.is_updating){
006              return;
007          }
008          var diffX = Math.abs(Accel.x) -
prevX;
009          if (diffX >= threshold) {
010              refreshQuote();
011          }
012              prevX = Math.abs(Accel.x);
013          }
014          , function(){}
015          , {frequency : 750}
016      );
017  }
```

We need to parse the incoming JSON response from the AJAX request into a readable format so that we can access specific nodes.

```
001  var request = new XMLHttpRequest();
002  request.open("GET", "file:///android_asset/
www/quotes.json");
003  request.onreadystatechange = function() {
004
005  if (request.readyState == 4) {
006      var data = JSON.parse(request.
responseText);
007      arrQuotes = data.quotes;
008      getQuote();
009  }
010
011  }
012  request.send();
```

We make use of jQuery's fantastic attribute selector abilities to easily reference the DOM elements for manipulation.

```
001  <blockquote id="quote"
class="quoteBlock"></blockquote>
002  <p rel="cite"></p>
003  <button id="refreshBtn" class="super button
pink">Show Another</button>
```

Run multiple JavaScript threads with Web Workers

Gen up on the Web Workers API and how it can maximise processing power

tools | tech | trends Text editor (eg Dreamweaver; not Microsoft Word) and a web browser **experts** Eric Freeman, Elisabeth Robson

On the CD
• manager.html
• worker.js

Browsers have always had a bit of a handicap in that all JavaScript code is run within one single thread. While this makes many tasks less error prone, it can result in an unresponsive user interface, and with today's new multi-core machines, it's a shame not to be able to use additional threads.

With HTML5, we now have a way to make use of JavaScript threads through the Web Workers API. In this two-part tutorial, we'll write a widget you can plug in to any page. The code for the widget is in a separate worker and runs in its own thread. This widget monitors a Twitter feed, and updates the main browser page when it receives new data. With Web Workers we can create a simple, self-contained worker that handles all the heavy lifting, leaving the main page JavaScript code to display the results and take care of everything else.

In part one we'll get acquainted with the Web Workers API, and then in part two we'll use a web worker to create a Twitter sentiment widget.

01 Web Workers overview

When using Web Workers, you split your code into two files: the main page (manager.html) and the worker (worker.js). The main page is in charge of creating the workers, sending them tasks to work on, getting results, and making any needed updates to the page. Once they're created, workers execute the code within the worker JavaScript file. Workers always send results back to the main page because they can't access the DOM directly.

02 Create the main page

We'll start with a simple page (manager.html) that has just one <div> element where we can display results we get from a worker (lines 10-11).

```
001 <!doctype html>
002 <head>
003 <meta charset="utf-8">
004 <title>Simple Web Workers</title>
005 <script>
006 // code to create a worker will go here
007 </script>
008 </head>
009 <body>
010     <div id="result">
011     </div>
```

> **"** With today's new multi-core machines, it's a shame not to be able to use additional threads **"**

```
012 </body>
013 </html>
```

03 Write a simple worker

Our first example is very straightforward. All the worker does is send a message to the main page that says 'hello main page'. Here's how you do that (in worker.js) – it's relatively easy:

```
001 postMessage("hello main page");
```

04 Employ a worker

To make a worker, you need to instantiate a worker object, which takes the file name of a worker JavaScript file. First, within the <script> element (in manager.html), create a new worker (line 1).

```
001 var aWorker = new Worker("worker.js");
```

05 Receive the worker's message

To start the process for receiving a message from a worker, you set up a handler in much the same way you set up other handlers in JavaScript: by assigning a handler function to the worker's onmessage property. So, create a new function – onWorkerMessage (line 4) – and assign it to the onmessage property of the worker (line 2).

```
001 var aWorker = new Worker("worker.js");
002 aWorker.onmessage = onWorkerMessage;
003
004 function onWorkerMessage(evt)
005 {
006   // message handling code will go here
007 }
```

06 Write the handler

Notice that the onWorkerMessage() function has an event parameter.

This parameter is an object with a property – data – which contains the message from the worker (lines 4-5). We're going to use the data property to update the result <div> with the data (line 6).

```
001  var aWorker = new Worker("worker.js");
002  aWorker.onmessage = onWorkerMessage;
003
004  function onWorkerMessage(evt) {
005      var data = evt.data;
006      document.getElementById("result").innerHTML = "Worker
said " + data;
007  }
```

07 Test your code

Load the manager.html file into your browser and you should see 'Worker said hello main page'. Note that in some browsers (like Chrome), you must serve the manager.html webpage using an HTTP server, not simply by loading the file into your browser – so make sure you are using a local server, or upload the files to an online server to test. This is a security restriction in some browsers, and fails silently (with a message to the console), so if you see nothing, check to make sure you're using a server.

08 Send a memo

You can also send a message from the main page to a worker. Use postMessage() to execute this (line 2).

```
001  var aWorker = new Worker("worker.js");
002  aWorker.postMessage("ping");
003  aWorker.onmessage = onWorkerMessage;
004
005  function onWorkerMessage(evt)
006  {
007      var data = evt.data;
008      document.getElementById("result").innerHTML = "Worker
said " + data;
009  }
```

09 Receiving the message

For the worker to receive the message from the main page, you need to add an onmessage handler in the worker (worker.js). Let's rework the worker code to do that. The worker first sets up the onmessage event handler (line 1), which is processed by the handleMessage() function. In the handleMessage() function, go and grab the data from the message (line 4), and run a test to see if the message says 'ping' (line 5). If so, respond with 'pong' – replacing 'hello main page' (line 6).

```
001  onmessage = handleMessage;
002
003  function handleMessage(evt) {
004      var data = evt.data;
005      if (data = "ping") {
006          postMessage("pong");
007      }
008  }
```

10 Test again

This time the worker will wait until it gets the message ping from the main page, and only then will it respond with its message: pong. Go right ahead and give this code a try. You should end up seeing 'Worker said pong' in the browser window if it has been set up correctly.

Non Worker

Canvas is running whilst an prime number finder runs - this will cause your browser to hang

Prime found: 3396091

[start worker]

HTML5 demos / @rem built this / view source
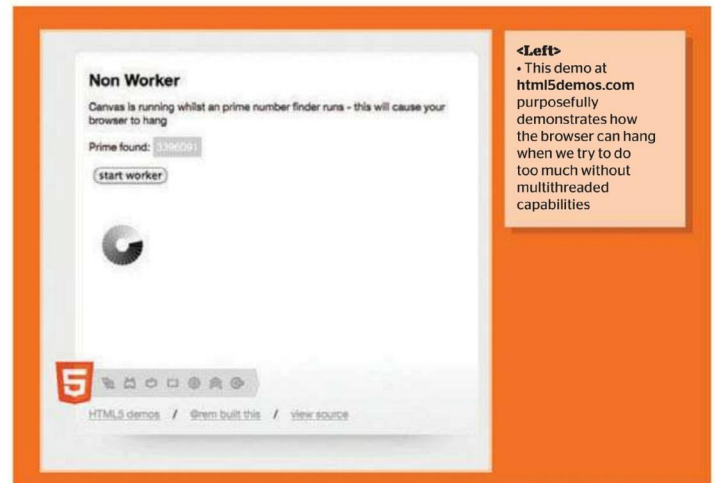
The traditional single thread

As we've established, up until now browsers have always been burdened by running JavaScript instructions within a single thread - something that could often result in a laggy or unresponsive experience.

Onload handler

Click handler

Validate form

Load data

Click handler

Validate form

Menu animation

Two files better than one

Using Web Workers requires you to split your code into two files, with a main HTML page being charged with spawning the workers, which then execute a worker .js file.

Making the leap into multi-threaded processing

The fairly revolutionary upgrade to HTML5 has brought with it the Web Workers API, which enables us to remove processes from the traditional single thread, to be processed within new threads running alongside.

```
Load data
```

| Onload handler | Click handler | Validate form | Click handler | Validate form | Update page | Menu animation |
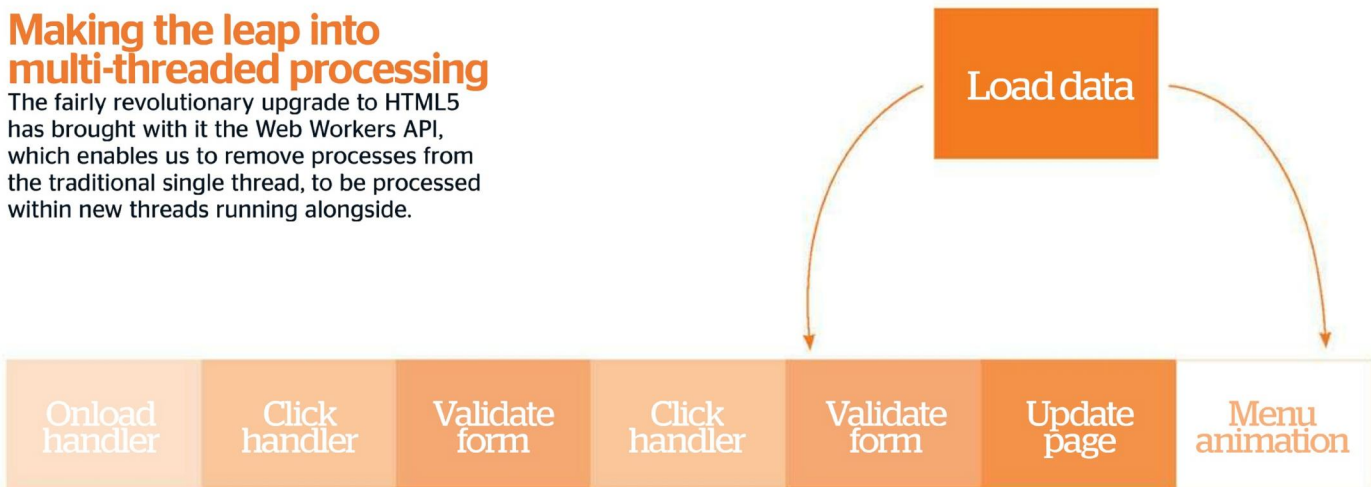
11 Add a timer

Say you want a worker to do something occasionally, like send a pong message every 2,500 milliseconds. You can use setTimer() within a worker just like in any JS code. Create a counter – count – to track the number of pongs in the worker (line 1). Then create a new function – sendMessage() – to handle posting a pong message back to the main page (lines 11-14). In that function we'll add a setTimeout() to call the sendMessage() function again in 2,500 milliseconds (line 13). Don't forget to replace the call to postMessage() in the handleMessage() function with a call to sendMessage() instead (line 7).

```
001  var count = 0;
002  onmessage = handleMessage;
003
004  function handleMessage(evt) {
005      var data = evt.data;
006      if (data = "ping") {
007          sendMessage();
008      }
009  }
010
011  function sendMessage() {
012      postMessage("pong " + count++);
013      setTimeout("sendMessage()", 2500);
014  }
```

12 Check your progress

Test your code again, by reloading manager.html. Now you should see a pong message from the worker, with a count in the message that increments every 2,500 milliseconds. The worker will continue sending pong messages as long as the page is active.

13 Send more complex messages

So far, all you've sent to and from your worker are strings, but you can also send objects to and from workers using JavaScript Object Notation (JSON). Change the worker (worker.js) to send an object to the main page, instead of a basic string. Send an object literal with two properties: a message property of pong, and a count property that holds the number of times the message has been sent (lines 12-13).

```
001  var count = 0;
002  onmessage = handleMessage;
003
004  function handleMessage(evt) {
005      var data = evt.data;
006      if (data.message = "ping") {
007          sendMessage();
008      }
009  }
010
011  function sendMessage() {
012      postMessage( { message: "pong", count: count }  )
013      count++;
014      setTimeout("sendMessage()", 2500);
015  }
```

14 Update the main page's message handler

Now that you're sending an object back to the main page, you need to update the code in the main page. In manager.html, use the object you're receiving from the worker to retrieve the message and count values (line 6). First, use the message property of the object to check to make sure you're receiving a pong message (line 7), and get the count of the message using the count property (line 9). Update your code to display the message and the count using the two different object properties (lines 8-9).

```
001  var aWorker = new Worker("worker.js");
002  aWorker.postMessage("ping");
003  aWorker.onmessage = onWorkerMessage;
004
005  function onWorkerMessage(evt) {
006      var data = evt.data;
007      if (data.message == "pong") {
008          document.getElementById("result").innerHTML = "Worker said "
009              + data.message + ", count is: " + data.count + "<br>";
010      }
011  }
```

to update the worker to handle this kind of message. First, update the worker (worker.js) to save the worker's id in a new variable id (lines 2, 7). Then, so that the main page knows which worker is responding, also update the object you're sending back to the main page in sendMessage() to include the worker number in the response (line 14).

```
001  var count = 0;
002  var id;
003  onmessage = handleMessage;
004
005  function handleMessage(evt) {
006      var data = evt.data;
007      id = data.id;
008      if (data.message = "ping") {
009          sendMessage();
010      }
011  }
012
013  function sendMessage() {
014      postMessage( {message: "pong", "count": count, "id": id } );
015      count++;
016      setTimeout("sendMessage()", 2500);
017  }
```

18 Update the main page

Edit the main page again (manager.html), to get the worker's id from the response and display it with the message and the count (line 13). You'll use this number to distinguish between the messages you're getting back from the various workers. Also change the line of code that modifies the innerHTML of the result <div> so that you're adding each new message to the <div> (rather than replacing the previous message) so you can see the results from each worker as they arrive (line 12).

```
001  var numWorkers = 3;
002  var workers = [];
003  for (var i = 0; i < numWorkers; i++) {
004      workers[i] = new Worker("worker.js");
005      workers[i].postMessage( { message: "ping", id: i } );
006      workers[i].onmessage = onWorkerMessage;
007  }
008
009  function onWorkerMessage(evt) {
010      var data = evt.data;
011      if (data.message == "pong") {
012          document.getElementById("result").innerHTML +=
013              "Worker " + data.id + " said "
014              + data.message + ", count is: " + data.count + "<br>";
015      }
016  }
```

19 Test your workers

Reload manager.html and test your workers. You should see messages from all your workers every 2,500 milliseconds. Notice that the messages from the workers don't necessarily come in any particular order, or in the same order each time. So, for example, if your workers are computing part of an image, then you'll need to include data in the response from the worker that tells the main page which part of the image that worker was working on, so the main page code can piece the various responses together correctly.

15 Time for another test

Test your code again. Now you should see a slightly different message from the worker, and behind the scenes you know that your main page and worker are now communicating with objects instead of just strings.

16 Create multiple workers

So far, we've used only one worker. And while one worker can be useful for many web applications (as you'll find out in part two of this tutorial series), you may also find that several workers can be useful when you have computationally intensive tasks. For example, multiple workers can be used for image processing, video or audio processing, generating or 'prefetching' data, updating a database, and much more. Update your code (manager.html) to create three workers (lines 1-7), and to send each a personalised message that consists of an object with two properties: a message property with the value ping, and an integer id, unique for each worker (line 5). We'll also store a reference to each worker in an array (line 2):

```
001  var numWorkers = 3;
002  var workers = [];
003  for (var i = 0; i < numWorkers; i++) {
004      workers[i] = new Worker("worker.js");
005      workers[i].postMessage( { message: "ping", id: i } );
006      workers[i].onmessage = onWorkerMessage;
007  }
008
009  function onWorkerMessage(evt) {
010      var data = evt.data;
011      if (data.message == "pong") {
012          document.getElementById("result").innerHTML = "Worker said "
013              + data.message + ", count is: " + data.count + "<br>";
014      }
015  }
```

17 Update the worker

Now you're sending a slightly different message to the worker: an object that includes both the worker's id and the message ping. So you need

Run multiple JavaScript threads with Web Workers: part 1

Try increasing the number of workers. However, you don't want to create too many; workers do require resources to create so for any application that uses workers, you'll want to balance the benefit of workers against the overhead of creating them and determine the optimal number for a given application. Using the number of cores you have in your machine as a basis for the number of workers is often a good starting point.

20 Handling errors

What do you do if your worker has an error? Well, you can tackle it with an error handler. Just like the message handler, you set up the error handler after you've created your worker. Edit manager.html and add an error handler for each worker in the loop where you set up the workers (line 7). Then add the function onWorkerError() (lines 19-23). The event object which is passed to the error-handling function includes some useful information to help you determine the source of the error in the worker, including the filename used to create the worker (line 21), the line number where the error occurred, as well as the error message (line 22).

```
001 var numWorkers = 3;
002 var workers = [];
003 for (var i = 0; i < numWorkers; i++) {
004     workers[i] = new Worker("worker.js");
005     workers[i].postMessage( { message: "ping", id: i } );
006     workers[i].onmessage = onWorkerMessage;
007     workers[i].onerror = onWorkerError;
008 }
009
010 function onWorkerMessage(evt) {
011     var data = evt.data;
012     if (data.message == "pong") {
013         document.getElementById("result").innerHTML +=
014             "Worker " + data.id + " said "
015             + data.message + ", count is: " + data.count +
"<br>";
016     }
017 }
```

```
018
019 function onWorkerError(evt) {
020     document.getElementById("result").innerHTML +=
021         "Error in worker " + evt.filename + " at line " +
022         evt.lineno + ": " + evt.message + "<br>";
023 }
```

21 Make a mistake

The easiest way to create an error in your worker so you can test your error handler function is to use a variable that doesn't exist. Edit worker.js, and create an error for worker 0. In handleMessage(), test to see if the worker's id is 0 (line 8). If it is, then create an error by setting a variable X to the value of a variable Y, which doesn't exist (line 9). This will create an error in the worker. The other workers will not create this error and will continue to run normally.

```
001 var count = 0;
002 var id;
003 onmessage = handleMessage;
004
005 function handleMessage(evt) {
006     var data = evt.data;
007     id = data.id;
008     if (id == 0) {
009         x = y;
010     }
011     if (data.message = "ping") {
012         sendMessage();
013     }
014 }
015
016 function sendMessage() {
017     postMessage( { message: "pong", "count": count, "id": id } );
018     count++;
019     setTimeout("sendMessage()", 2500);
020 }
```

Design for a better web

Add creativity and flair to your online projects with an essential range of training products for hobbyists and professionals

Bookazines

FREE CD

450 tips and techniques inside

WEB DESIGN
Creative guides to master the web

Volume Six

EXPERT TUTORIALS TO DESIGN NEXT-GENERATION WEBSITES

Web Design Vol. 6
Creative guides for designing next-gen websites. In-depth features to bring you up to speed on all the hottest technologies. Step-by-step tutorials showing you how to design amazing websites using CSS3, HTML5, Flash, Photoshop and more.
£14.99

Back issues

web **designer**
Flash AJAX
WEB CODES CLASH

web **designer**
BLOGGER KING

web **designer**
WORLD'S WORST WEBSITES

web **designer**
CLEANER CODE
CSS

£12.99
£6.49 with code

Beginner's guide to...
WORDPRESS
W
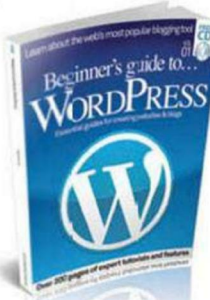
Beginner's Guide To WordPress Vol. 1
The WordPress Genius Guide offers in-depth features and tutorials on the world's most popular blogging platform.
£12.99

Web Design

Web Design Vol. 5
This book provides a comprehensive collection of visual lessons on the essential techniques needed to create awe-inspiring websites.
£6.49 with code "DOTCOM"

DVDs

DIGITAL EDITION
Web Design
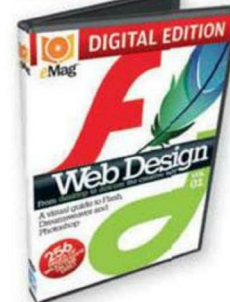
Web Design: From Desktop to Dotcom eMag Vol. 1
A visual guide to Flash, Dreamweaver and Photoshop. 256 pages of walkthroughs on one disc! A project-based workout for any webslinger! Master tools, receive technical advice and read inspirational features.
£9.99

web **designer** eMag

Web Designer eMag Vol. 1
Enjoy 21 issues of Web Designer on one interactive disc. Over 1800 pages of expert tutorials, case studies & features. Plus bonus tutorial workshop files included.
£9.99

£19.99
Web Designer DVDs are now 1/2 price...
£9.99

web **designer** eMag
824 pages of cutting-edge trends and techniques
FREE 256

Web Designer eMag Vol. 2
A complete archive of Web Designer magazine issues 137-148. Over 824 pages of cutting-edge trends and techniques! Free – 256 page Web Design Vol. 03 bookazine included in pdf format.
£9.99

Become an ImagineShop customer and leave reviews of your favourite products.

Hurry, stock is limited!

Order online **imagineshop**.co.uk

ip IMAGINE PUBLISHING

Email eshop@imagine-publishing.co.uk
Twitter @imagineshopuk

| MAGAZINES | BOOKS | DVDs | DOWNLOADS | GIFTS |

01 www.danilogiagnoli.com

02 www.opentecheng.com

Danilo Giagnoli

web www.danilogiagnoli.com

Current role	Graphic/web designer at Opentech ENG
Education	Degree in Image and Communication at the University of L'Aquila; self-taught web designer
Expertise	HTML5, XHTML, CSS, JavaScript, jQuery, jQTouch, Dreamweaver, Edge, Fireworks, Photoshop, Inkscape, brand design
Clients	UniCredit, ING Direct, La Repubblica, Webank
Twitter	@DGiagnoli

Danilo Giagnoli is a web designer from Italy's capital, Rome. He started creating websites nine years ago on a freelance basis and his knowledge comes from a passion and love for design. His personal site has already received several awards and accolades, including a review in an Italian web-design magazine and some CSS Design Awards.

Currently Giagnoli works at Opentech ENG, a major company operating in the mobile industry. When he's developing sites he takes care of every aspect, from information architecture, prototyping, concepting, interaction design through to implementation.

To inspire his work, he usually looks up CSS showcase galleries (like CSS Remix), magazines, advertising, product packaging and architecture-focused books.

Giagnoli strongly believes in web standards and in semantic markup, and he is convinced that we are living in a really exciting and creative time. HTML5, CSS3, responsive web design, mobile and web fonts all give us the opportunity to create solutions that just a few years ago were impossible.

To stay updated he follows useful sites like http://stuffandnonsense.co.uk/blog, http://css-tricks.com, http://webdesignerwall.com and Twitter. He also loves to regularly update his blog (www.danilogiagnoli.com/blog).

Giagnoli says: "I like communicating through my blog as I can receive feedback on my own topics from readers that are interested in what I am writing about." He also enjoys mountain biking, watching movies and spending time with his sweetheart, Alessia.

03 www.mayaprogettazioni.com

04 https://m.ingdirect.it/mobilewebapp

01
Giagnoli constantly adds all of his best web, graphic and illustrative creations to his personal site. The portfolio is based on a vertical layout in order to offer a quicker, more immediate view of his work.

02
For the Opentech ENG website, Giagnoli took care of the graphic concept, user experience and HTML/CSS implementation. Once completed, the site was optimised for mobile devices too.

03
Maya is a young construction company. The layout of its web presence is characterised by a very minimalist and clean style. The tools used to build it include HTML, CSS and jQuery.

04
ING Direct is a famous bank with offices and customers all over the world. For this big client, Giagnoli took care of the HTML/CSS implementation of the mobile site as well as some tablet optimisations.

Tim Smith (aka My Poor Brain)
web www.mypoorbrain.com

Current role	Senior designer
Education	BA (Hons) degree in Graphic Design & Professional Media (University of Gloucestershire)
Expertise	Flash, Photoshop, brand design, illustration, animation
Clients	Orange, PlayStation, Nokia, *Computer Arts*, Evian, Tate, The British Museum, Macmillan Cancer Support, Game, Symantec
Twitter	@mypoorbrain

Tim Smith is a graphic designer, illustrator and web designer from North Wales, now residing in the creative capital that is London. Smith started web design fairly early on in his life. From drawing logos at the age of four to creating pixel work on his older brother's Amiga 500+ and MS Paint, his first venture into web design was at the age of 14, designing and building a Feeder fansite. His efforts caught the attention of the band themselves, prompting Feeder to urge fans to visit his site rather than the official one!

After graduating from the University of Gloucestershire, Smith soon found himself in London doing what he does best: working on a huge variety of projects. He is well known for the variety in his work and has designed many web and interactive pieces, from DVD interactives for Sony PlayStation, to websites for Nokia, to online games for The British Museum, to Facebook apps for Symantec and beyond. This is as well as print and branding work for Orange, Macmillan Cancer Support, Evian and Game among others, for which he has received recognition from the likes of the Webbys, Design Week and the FWA.

It's also in his eclectic design style that Smith demonstrates his versatility – he prefers to let the brief and concept dictate the aesthetic and design, rather than a personal repertoire of styles, and herein lies this web creative's inspiration. The challenge of the brief and the conceptual solution is what ultimately drives all of Smith's work.

01 http://tinyurl.com/sm-drench

02 www.fansite.so

03 http://tinyurl.com/Sh-the-d

05 http://tinyurl.com/bm-time
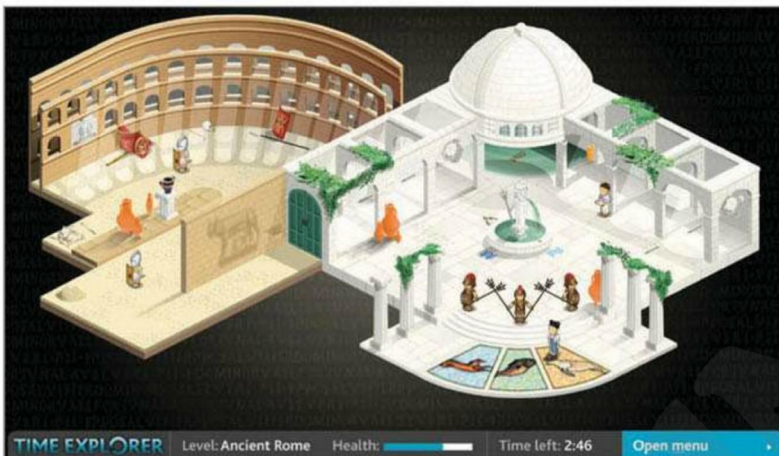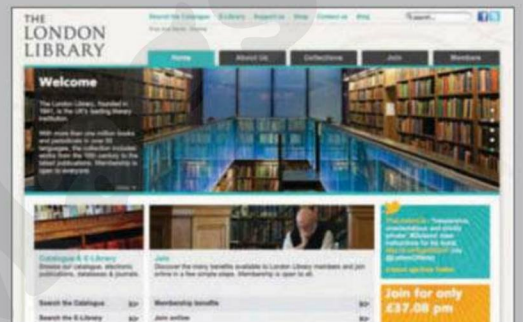
04 www.londonlibrary.co.uk

01 Drench may be the first isometric 3D tower defence game ever. The premise of the game is to capture fresh water and keep the rising population alive.

02 FanSite is a contemporary, digitised equivalent of a fan scrapbook. It's a social network for fans to get together, debate and swap favourite content.

03 The iPhone version of a cult-classic playground card game. It includes fully customisable rule sets and can host up to three players.

04 The world's largest independent library required a new online presence. It balances modern design and photography to show off the institution's rich heritage.

05 Travel back in time to three historic civilisations in order to rescue artefacts from natural disasters. A cutting-edge Flash game with an educational message.

01 www.hullcityosc.com

02 www.shackeltons.co.uk

03 www.team-wear.co.uk

Nicky Lock
web http://nicky-lock.com

Current role	Digital designer
Education	BA Multimedia and Communication Design (Sheffield Hallam University)
Expertise	Graphics, logo design, web design, Flash, CMS, animation
Clients	Hull City Official Supporters' Club, Team Wear Sports Clothing, JDF Writing, Mountain Rescue
Twitter	@nickthedesigner

Nicky Lock is a digital designer based in Sheffield, UK, with over four years' experience. Having studied communication design at uni, he now works for the marketing and branding agency Ledgard Jepson.

Lock began his career as an in-house web designer for an online retailer combined with being freelance, working with a variety of clients in the UK on digital-based projects from websites and branding to mobile ventures.

He specialises in UX/UIs, creative web design and frontend development. Having recently made the next step in his career by moving to an agency, he looks forward to tackling some new challenges in the areas of mobile apps and responsive web design.

At school, Lock's strongest subjects were art and IT and it was his fascination with combining the two that led him down the career path of a digital-based designer. Eager to share his skills and knowledge with others, he maintains a strong social presence online in order to improve the quality of both his own work and that of others. He encourages all designers to get their voice out into the creative community and show people what projects they're currently working on.

Looking forward, with much anticipation of evolving technologies like HTML5 and CSS3, Lock hopes to push the boundaries of web design even further and secure himself a place as a leading figure in the field.

04 www.higear.uk.com

01
This website for a local football club combines improved style and imagery with an online shop feature to give Hull City OSC a fresh and informative online presence for all the team's fans.

02
Using elegant typography and amazing imagery of their work alongside a wooden texture, Lock built an online presence for a hugely talented joinery that's based in Yorkshire.

03
With a full-background image slider showing product photography, this WordPress site uses colour and web fonts to give structure and a professional feel to a clothing company's portfolio.

04
This eCommerce site design for a bestselling outdoor clothing and equipment company uses an in-keeping organic colour scheme and stylistic layout to represent the values of the brand and its range of products.

14-PAGE DEVELOPER SECTION
Features and tutorials dedicated to web coding

CSS3 STYLE SECRETS

TOP TIPS FOR APPLYING A NEXT-GEN MAKEOVER TO YOUR PAGES WITH THE TECHNIQUES YOU NEED TO DITCH PHOTOSHOP FOREVER

KINETIC MOTION
Build an experimental solar system animation with complex HTML5 and CSS3

PRO FILE: AIRLOCK
Brand strategies, creative development, UX planning and more from this London agency

HTML5 GAME DESIGN
Create a sophisticated platform game combining progressive markup with EaselJS

COFFEESCRIPT
Enhancing JavaScript's readability and adding brand-new features with this buzz compiler
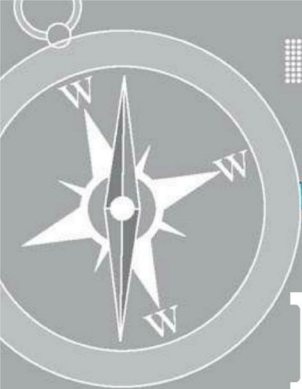
Hotbeds of worldwide web design

Helsinki

Explore the official World Design Capital of 2012

1 Activeark
www.activeark.com

Activeark is a full-service agency built for today's digital society. The company consists of 80 design-driven digital professionals located in three locations around the world and headquartered in Helsinki. There is tough competition among Helsinki's marketing agencies, but Activeark's consistent drive for innovation has made it one of the key players in the city's digital design scene.
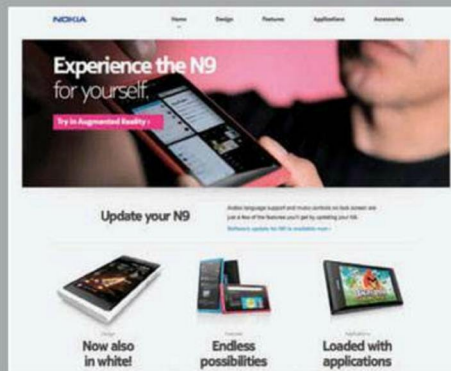
Finnish design is probably best known for its architecture, with famous masters like Alvar Aalto or Eero Saarinen, or the traditional Scandinavian style of household brands like Marimekko prints, Iittala glassware or Artek furniture. But right here in the 21st century, there's a whole lot more to Finnish design.

Helsinki – official World Design Capital 2012 – has an interesting year ahead of it. Art and design is literally filling the streets and thousands of creative-minded tourists are arriving to see what the Finns have to offer the design world. For a creative, this is a fantastic year to be part of this city. With a multitude of workshops, exhibitions, architectural projects and competitions, the list of what to do and see in this coming festival year is endless.

For a digital designer the year's main theme – 'Better living through design' – makes sense. Who wouldn't want to create a service or a platform that really makes a tangible difference to people in their everyday lives? This is what WDC Helsinki 12 is all about – inspiring people to think and to create.

Take a look at a few of Helsinki's creative agencies showcased here to get an idea of the kind of talent on display for WDC Helsinki 12. What do you get when you bring together digital professionals, award-winning identity designers, eccentric illustrators, leading photography talent and skilled graphic designers? A truly inspiring mix that's a far cry from traditional Scandinavian minimalism.

Beyond the more 'traditional' design establishments, Helsinki is bustling with small independent design agencies, ad houses, studios, illustrators, fashion designers, freelancers and artistic collectives working in all fields, both for Finnish clients and globally.

It's this bricolage of creative talent and wealth of innovation that gets us so excited about WDC Helsinki 12. Sample some of the hottest featured agencies here now.

WDC 2012 is all about inspiring people to think and to create

2 Kokoro & Moi
www.kokoromoi.com

Kokoro & Moi is a multidisciplinary design agency specialising in brand identity and development, creative direction, art direction, plus graphic and interactive design. Drawing on unexpected combinations, Kokoro & Moi is renowned for its innovative and expressive approach to the entire modern media palette.

3 Agent Pekka
www.agentpekka.com

Agent Pekka is a creative consultancy firm specialising in commercial artist management. Its comprehensive roster of creative talent includes illustrators, CGI artists and moving-image designers as well as three creative agencies, bringing together a broad range of expertise. The selection of illustrators combines both emerging talents and internationally established artists like Jesse Auersalo, Inka Järvinen and Rami Niemi.

4 Luxus
http://luxus.fi

Luxus started out back in 2001 with just five employees. Over the intervening years the agency has grown into the number one digital agency in Finland and now employs over 100 staff. They are passionate about music, photography and fishing and boast an accomplished crew of creatives, producers and techies.

5 Tsto
http://tsto.org

Tsto is a creative agency founded by six designers. Tsto's graphic design professionals are specialised in visualising their ideas to help clients define and communicate their identity and message. Tsto marries design thinking with a craftsman's can-do attitude and looks to collaborate rather than compete with other agencies.

Did you know that Finns drink more cups, mugs and shots of coffee per capita than any other nation in the world?

Recommended hotspots

Minna Hiltunen,
lead creative designer
Punavuori design district
http://bit.ly/x93CDH

This is a bohemian neighbourhood of Helsinki creatives. Vintage shops, cosy cafés, laidback bars and an urban atmosphere create the perfect environment for design studios and agencies.

Tuukka Sassi,
senior service designer
WDC Helsinki 2012
http://wdchelsinki2012.fi

Official website for WDC Helsinki 2012, this is the place to go to find out what's happening during the Design Capital year. The site lists hundreds of events both in Helsinki and nearby cities.

Taneli Sippola,
lead creative designer
Merihaka
en.wikipedia.org/wiki/Merihaka

It's a controversial monumental structure that has a strong timestamp on it, but as a designer you can only admire the systematic minimalism and functionalities in this 'metropolis of the future'.

Talent directory

Whether you need to hire web-design services or just want to locate the trade's best creative teams, our chart can help…

2Advanced Studios
www.2advanced.com

City Aliso Viejo
Telephone 1 949 521 7000
Contact info@2advanced.com
Clients Adobe, Bacardi, Diesel

Country USA
Twitter ID 2advanced

Established in 1999, 2Advanced is an award-winning interactive agency dedicated to pushing the creative boundaries of design and technology to deliver high-impact, rich media solutions. It boasts an exceptional reputation in Adobe Flash design and offers a host of services including interactive design, CMS solutions, eCommerce, motion graphics, 3D illustration and video production.

NAME	URL	CITY	COUNTRY	TELEPHONE	CONTACT	TWITTER ID	CLIENTS
Addictivity	www.addictivity.com	London	UK	0845 838 6718	hello@addictivity.com	addictivity	Hitachi, Kickers, Motorola
Agency Republic	www.agencyrepublic.com	London	UK	44 (020 7942 0000	chat@agencyrepublic.com	agencyrepublic	adidas, EA, BBC, BP, O2
AKQA	www.akqa.com	London	UK	(0)20 7780 4786	info@akqa.com	AKQA	Coca-Cola, Virgin, McDonald's, Nike, Visa, Xbox
Angel London	www.angellondon.co.uk	London	UK	0845 468 0959	info@angellondon.co.uk	angellondon	Sony Pictures, Chelsea FC, MTV, BT
AWP New York	www.awpny.com	New York	USA	607 277 2757	www.awpny.com/pages/feedback	AWPNY	Accufab, Cornell University

Azexis
www.azexis.com

City Ipswich
Telephone 01473 694500
Contact sales@azexis.com
Clients BBC, Kia, Business Link, LSC

Country UK
Twitter ID Azexis

With a heritage in design and development since 1999, Azexis is an interactive agency with a talented team of specialists all sharing a single drive: to produce the best design, web and mobile solutions encompassing extraordinary functionality and creative design. Azexis offers a diverse range of services and supports an impressive array of clients worldwide.

NAME	URL	CITY	COUNTRY	TELEPHONE	CONTACT	TWITTER ID	CLIENTS
BBH Global	www.bartleboglehegarty.com	London	UK	(0)20 7734 1677	richard.stainer@bbh.co.uk	bbhlondon	Axe, Baileys, LG, Sprite, Heineken
BGT Partners	www.bgtpartners.com	Chicago, IL	USA	1 888 282 2487	bgtla@bgtpartners.com	diclarke	Burger King, FedEx, Walt Disney
Big Youth	www.bigyouth.fr	Paris	France	171183100	contact@bigyouth.fr	N/A	AOL, Sony Ericsson, Nivea, Orange
Big Spaceship	www.bigspaceship.com	New York	USA	1 718 222 0281	biz@bigspaceship.com	bigspaceship	Sony Pictures, Microsoft, Victoria's Secret
Birth	http://birth.se	Stockholm	Sweden	46 (0)8 411 5775	hello@birth.se	N/A	Red Bull, DDB Paris, McDonald's, DHL, Telenor
Blitz	www.blitzagency.com	Santa Monica, CA	USA	310 551 0200	www.blitzagency.com/contact.aspx	BLITZagency	Activision, Google, Starbucks
BlueInkAgency.com	www.blueinkagency.com	Grimsby	UK	0790 558 8144	enquiries@blueinkagency.com	BlueInkAgency	Hein Gericke, Nicholas Deakins, Crane Gregory
Breathe 365	www.breathe365.com	Liverpool	UK	0151 324 0188	info@breathe365.com	N/A	Saatchi & Saatchi, Siemens
BURN	www.burnmarketing.com	Richmond, Surrey	UK	020 8332 7204	www.burnmarketing.com/contact-us	BURNMARKETING	lastminute.com, eBay, O2, One Water
Carsonified	http://carsonified.com	Bath	UK	01225 324 980	hello@carsonified.com	carsonified	N/A
Catch Digital	www.catchdigital.com	London	UK	0207 494 3554	hello@catchdigital.com	catch_digital	Orange, Hellman's, Crest Nicholson, Altitude
Chunk	www.chunk.co.uk	Glasgow	UK	0141 353 0876	hello@chunk.co.uk	chunktwits	Kronenbourg, Cadbury's, Volvo
CHIPS	http://chips-ny.com	New York	USA	N/A	office@chips-ny.com	chipsny	Top Magazine, Joffrey, d'employ, Matt Creed
Ciplex	www.ciplex.com	Los Angeles, CA	USA	1310 461 0330	www.ciplex.com/contact_us.html	ciplex	Upper Deck, Von Dutch
Clearleft	http://clearleft.com	Brighton	UK	0845 838 6163	info@clearleft.com	clearleft	WWF, Universal Networks, Mozilla
Code23	www.code23.com	Reading	UK	44 (0) 118 9306717	www.code23.com/contact/	code23	Oasis James, Sony Style, Adams Estates
Content Formula	www.contentformula.com	London	UK	0207 3816671	moreinfo@contentformula.com	contentformula	Johnson & Johnson, Kraft, Bausch & Lomb
Coolpink	www.coolpink.net	Leeds	UK	44 113 201 8290	discovermore@coolpink.net	coolpink	Asda, NHS, Warner Music, Halfords
Cre8 New Media	www.cre8newmedia.com	Ruislip	UK	01707 394 962	info@cre8newmedia.com	N/A	Agenda Recruitment, Ocean Group
Creare Design	www.webdesigncreare.co.uk	Leicester	UK	01455 88 37 00	enquiries@creare.co.uk	crearegroup	Nicky Clarke, Life Music Foundation, Iveco

Critical Mass
www.criticalmass.com

City London
Telephone +44 20 8735 8750
Contact newbusiness.europe@criticalmass.com
Clients Rolex, NASA, HP, adidas

Country UK
Twitter ID criticalmass

CRITICALMASS

A leading, full-service digital agency with offices in Toronto, New York and London, Critical Mass offers 15 years' industry experience, having been founded in 1996. Services include project planning, experience design, content marketing, social media strategies and rich technology development – to name just a few.

NAME	URL	CITY	COUNTRY	TELEPHONE	CONTACT	TWITTER ID	CLIENTS
d2x	www.d2x.co.uk	Basingstoke	UK	07512 076360	info@d2x.co.uk	d2x_co_uk	Altro, Walkingworld
Dare London	www.daredigital.com	London	UK	44 (0)203 451 9101	london@thisisdare.com	thisisdare	Sony Ericsson, Hovis, Tetley, Penguin
Delete	www.deletelondon.com	London	UK	0203 004 7155	info@deletelondon.com	DeleteLondon	Red Bull, Decca Records
Dizzain	www.dizzain.com	New York	USA	44 (20) 3290 9941	mail@dizzain.com	dizzain	Webhuset, Financial Times, HIS Textiles, QxLab
Digital Jigsaw	www3.digitaljigsaw.com	London	UK	0870 235 4042	info@digital-jigsaw.com	digitaljigsaw	PepsiCo, Walkers, Budweiser
Digital Marmalade	www.digitalmarmalade.co.uk	Beckenham, Kent	UK	44 (0)20 8249 5655	info@digitalmarmalade.co.uk	N/A	Monarch, Yell.com, Trevor Sorbie
Disturb Media	www.disturbmedia.com	London	UK	020 7138 3569	contact@disturbmedia.com	disturbmedia	Swatch/MTV, Hertz
Domani Studios	http://domanistudios.com	New York	USA	312 850 2919	biz@domanistudios.com	domanistudios	Sheraton, Porsche, VW
DRIFTLAB	www.driftlab.com	Atlanta, GA	USA	678 225 4351	howdy@driftlab.com	driftlab	General Motors, Ford, Sony
eckhoCreative	www.eckhocreative.com	Bristol	UK	01179 070273	info@eckhocreative.com	eckhoCreative	Rocket Music Management, Nexus TV, ITV, CEF
Fi	www.f-i.com	New York	USA	212 9415220	interact@f-i.com	F_I	Google, HTC, BBC, Fox, Porsche, Burton, EA
Firstborn	www.firstborn.com	New York	USA	212 581 1100	info@fborn.com	firstborn_nyc	Apple, Doritos, Pepsi, Wrigley
Fluid Creativity	www.fluidcreativity.co.uk	Hyde, Cheshire	UK	0845 6588 373	info@fluidcreativity.co.uk	fluidcreativity	American Soda, Brother, NHS
Form	www.form.uk.com	London	UK	020 7014 1430	studio@form.uk.com	N/A	BBC, VH1, Cutty Sark Whisky
Glass Slipper	www.glassslipperinteractive.com	Banbury	UK	01295 724568	glassslipperinteractive.com/contact.php	dpaultaylor	Nexus Interiors, bell systemz
glue Isobar	www.glueisobar.com	London	UK	020 7739 2345	md@glueisobar.com	glueisobar	Oasis, The Sun
GR/DD	www.grdd.co.uk	London	UK	020 7729 4222	mail@grdd.co.uk	GRDD_London	PlayStation, Warner Bros, Symantec, Nokia
Guerilla	www.guerilla.co.uk	Newcastle	UK	4401912619799	enquiries@guerilla.co.uk	we_are_guerilla	Network One, Green Metropolis
Hi-ReS!	www.hi-res.net	London	UK	0207 729 3090	info@hi-res.net	N/A	Chanel, Dolce & Gabbana, Nike

Featured agency: 2Advanced Studios www.2advanced.com

A closer look at what each listed agency has to offer the community

2Advanced Studios is a small Californian agency founded in early 1999, by Tony Novak, Eric Jordan and John Carroll. Over the past 12 years the company has provided a wide range of services from its core offering of web design and development, working with an impressive list of clients. In previous years where Flash work (comprising motion graphics and ActionScript development) accounted for more than 95 per cent of 2Advanced's projects, the recent demand for HTML5 and CSS3 has seen Flash-based work drop to below 30 per cent of the agency's output.

NAME	URL	CITY	COUNTRY	TELEPHONE	EMAIL	TWITTER ID	CLIENTS
JUXT Interactive	www.juxtinteractive.com	San Francisco, CA	USA	949 610 0800	info@juxtinteractive.com	juxtinteractive	Sprite, Cisco, American Airlines, BMW, Adobe
Kokokaka	www.kokokaka.com	Gothenburg	Sweden	031 711 54 50	contact@kokokaka.com	KOKOKAKA	Burberry, IKEA, Wrangler
Kudos Web Design	www.kudoswebdesign.co.uk	Manchester	UK	0161 236 5585	info@kudoswebdesign.co.uk	N/A	Tetra Pak, Taboo
KURO	www.kuro.la	Long Beach, CA	USA	562 494 5070	projects@kuro.la	N/A	N/A
Liquid Light	www.liquidlight.co.uk	Brighton	UK	08456 58 88 35	info@liquidlight.co.uk	liquidlightuk	Antwords, Charcol, Gifford
Lonely Duck	www.lonelyduck.se	Stockholm	Sweden	46 70 689 96 02	hello@lonelyduck.se	LonelyDuckSTHLM	Puma, New Balance, Bentley
Lounge Lizard	www.loungelizard.com	New York	USA	1-888-444-0110	info@loungelizard.com	kenbraun	America Online, Disney Interactive
Lyrical Media	www.lyricalmedia.com	Worthing	UK	0208 123 1 567	lyricalmedia.com/contact	LyricalMedia	Virgin Atlantic, Ubisoft

Mutado
mutado.com

City Milan **Country** Italy
Telephone 39 02 34592733 **Twitter ID** mutado
Contact hello@mutado.com
Clients Vodafone, Yahoo!, Nike

Mutado®

Mutado was formed in 2004 by Lorenzo Manfredi and Mauro Gatti: a developer and a graphic designer whose paths crossed over a decade ago. During the last eight years, it has grown into a solid independent digital agency, headquartered in Milan. The company's name relates to Mutado's constant mutation, adaption and transformation.

Next Digital	www.nextdigital.com	Sydney	Australia	61 2 9336 7500	sydney@nextdigital.com	nextdigital	Brisbane Water, Qantas
Night Agency	www.nightagency.com	New York	USA	212 431 1945	info@nightagency.com	nightagency	Champion, ESPN, MTV, 20th Century Fox
North Kingdom	www.northkingdom.com	Skellefteå	Sweden	020 7749 5353	info@northkingdom.com	NorthKingdom	Coca-Cola, Vodafone, adidas
Nzime	www.nzime.com	Nottingham	UK	0115 950 9720	people@nzime.com	nzime	Elastoplast, Microsoft Game Studios, Knox
OMdeSIGN	www.omdesign.co.uk	London	UK	0208 248 2712	contact@omdesign.co.uk	omdesign	City of London, Loft King, The Hokey Cokey Man
Orange Bus	www.orangebus.co.uk	Newcastle	UK	0191 241 3703	stuff@orangebus.co.uk	@orangebus	Barclays, Sage Pearson
Pod1	www.pod1.com	London	UK	020 7524 7660	www.pod1.com/contact-us	Pod1	Tesco, Shell, Kenwood
POKE London	www.pokelondon.com	London	UK	2077495353	www.pokelondon.com/contact	pokelondon	Skype, Orange, Yahoo!
Razorfish	www.razorfish.com	Seattle, WA	USA	1 206 816 8800	www.razorfish.com/#/reach-us/email-us	Razorfish	adidas, Audi, Dell, O2
Reactive	www.reactive.com	London	UK	44 (0)20 7550 8200	uk.enquiries@reactive.com	reactivemedia	Lee Jeans, British Airways, NEC

Resn
www.resn.co.nz

City Wellington **Country** New Zealand
Telephone +64 4 385 0705 **Twitter ID** resn_has_no_i
Contact newbusiness@resn.co.nz
Clients Toyota Racing, BlackBerry, MTV

Resn

Independently owned and based in New Zealand's capital city, Resn is one of Australasia's most-recognised agencies, having scooped FWA, Webby, SXSW and Pixel awards. Founded in 2005 by Steve Le Marquand and Rik Campbell, the team is known for an innovative web and interactive portfolio of projects for a high-profile client list.

SimpleBits	http://simplebits.com	Salem, MA	USA	N/A	contact@simplebits.com	simplebits	MTV, AIGA, Google, ESPN
Soak Digital	www.soak.co.uk	Norwich	UK	01603 630631	www.soak.co.uk/contact-us.php	SoakDigital	Axa, Harley Davidson, UEA, Archant
Solid State Group	www.solidstategroup.com	London	UK	0207 613 7220	info@solidstategroup.com	solidstategroup	Sony, Shell, Amnesty International
Striker Digital	www.striker-digital.com	Prague	Czech Rep.	+42 (0)257 312 707	hello@striker-digital.com	strikerdigital	Mullermilch, STV, BP
Sweden Unlimited	www.swedenunlimited.com	New York	USA	212 941 5904	info@swedenunlimited.com	swedenunlimited	Vera Wang
TAMBA Internet	www.tamba.co.uk	London	UK	0203 008 5110	info@tamba.co.uk	TAMBA_Internet	Carphone Warehouse, EMI
Tokyo Digital	www.tokyodigital.co.uk	Liverpool	UK	0151 709 9524	gareth.martin@tokyodigital.co.uk	TokyoAgency	GWR, Letterpool, MOBO, Samsung
Tomato	www.tomato.co.uk	London	UK	0207 490 2599	info@tomato.co.uk	N/A	Citroën, Ford Ka , dunhill
Tribal DDB LA	http://tribalddb.com	San Francisco, CA	USA	415 732 3600	mike.parker@us.tribalddb.com	TribalDDB	Hasbro, McAfee, McDonald's
ultranoir	www.ultranoir.com	Paris	France	014 271 4836	online form	ultranoir	eBay, HSBC, Peugeot, Lacoste, Thomas Cook

Underwired Amaze
www.underwired.com

City London **Country** UK
Telephone 020 7060 0400 **Twitter ID** underwired
Contact hello@underwired.com
Clients McCain Foods, Peugeot, Virgin, ASICS, NSPCC

underwired

Now part of the Amaze network of companies after a 2009 merger, Underwired was founded as an agency 15 years ago. Started by Ruth Brecher, Jason Holland and Felix Velarde, it has become synonymous with customer engagement campaigns spanning websites, email newsletters, mobile applications and social media.

(untitled)	www.untitledlondon.com	London	UK	020 7871 7939	fanmail@untitledlondon.com	untitledlondon	Hasbro, Spar, Lloyds TSB, Sky, Sony
The Value Engineers	www.thevalueengineers.com	Beaconsfield	UK	01494 680999	info@thevalueengineers.com	ValueEngineers	BP, Orange, Sony
Vibe Creative	www.vibecreative.co.uk	Shrewsbury	UK	0845 263 7989	studio@vibecreative.co.uk	vibecreative	E&J Jewellers, Somerset Housing
Vivid Creative	www.vividcreative.com	Sheffield	UK	0114 261 7060	info@vividcreative.com	Vivid_Creative	N/A
Webheads	www.webheads.co.uk	London	UK	0207 287 7060	info@webheads.co.uk	webheadsdigital	BBC, NHS, Texaco
Wevio	www.wevio.com	New York	USA	1 212 537 9432	contact@wevio.co.uk	Wevio	Samsung, IBM, Verizon
Wickedweb	www.wickedweb.co.uk	London	UK	020 7183 4999	www.wickedweb.co.uk/contact	InsideWickedweb	Nectar, Kurt Geiger, NSPCC
Wieden+Kennedy	www.wkamst.com	Amsterdam	Holland	31.20.7126.500	lee.newman@wk.com	WKAmsterdam	EA, Heineken, Nike
Your Majesty	www.your-majesty.com	New York	USA	212 685 3427	info@your-majesty.com	yourmajestyco	Hyundai, Cisco, Yahoo!
The Zen Agency	www.thezenagency.com	Glasgow	UK	0141 229 1333	online form	zenagency	The HALO Trust, Mackie's, Hilton

Let **Web Designer** unleash your creative side!

Subscribe now and
SAVE 30%

The only magazine you need to design and develop stunning websites

- Dreamweaver, XHTML & CSS
- WordPress & microblogging
- Photoshop & Illustrator
- Flash & ActionScript
- Plus site showcases
& exclusive interviews!

Hosting listings

Keep an eye on the latest packages and deals with our comprehensive list of service providers

Got a deal you think we should list?

Whether you're a hosting firm keen to promote your products or a happy customer who wants a favourite provider to be listed, drop us a line with the details!

webdesigner@imagine-publishing.co.uk

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER YEAR	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
1 & 1 Internet Ltd. www.1and1.co.uk	1&1 Starter (Linux)	0844 335 12 11	£29.88	5GB	Unlimited	1,000	✓	✗	✓	✓	✓	✓	✓	✗
	1&1 Standard (Linux)	0844 335 12 11	£59.88	50GB	Unlimited	3,000	✓	✗	✓	✓	✓	✓	✓	✗
	1&1 Standard (MS)	0844 335 12 11	£71.88	50GB	Unlimited	3,000	✓	✗	✓	✓	✓	✓	✓	✗
	1&1 Unlimited (Linux)	0844 335 12 11	£83.88	Unlimited	Unlimited	5,000	✓	✗	✓	✓	✓	✓	✓	✗
	1&1 Unlimited (MS)	0844 335 12 11	£107.88	Unlimited	Unlimited	5,000	✓	✗	✓	✓	✓	✓	✓	✗
	1&1 Business (Linux)	0844 335 12 11	£119.88	Unlimited	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✗
	1&1 Business (MS)	0844 335 12 11	£155.88	Unlimited	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✗
111WebHost.com http://111webhost.com	Unlimited Web Hosting Pack	N/A	£60	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✗	✓	✓	✗
	Starter Web Hosting Pack	N/A	£30	5GB	2GB	100	✓	✓	✓	✓	✗	✓	✓	✗
	Budget Web Hosting Pack	N/A	£12	1GB	1GB	5	✓	✗	✓	✓	✗	✓	✓	✗
	WordPress Web Hosting Pack	N/A	£24	5GB	2GB	100	✓	✓	✓	✓	✗	✓	✓	✗
	Drupal Web Hosting Pack	N/A	£24	5GB	2GB	100	✓	✓	✓	✓	✗	✓	✓	✗
	Joomla! Web Hosting Pack	N/A	£24	5GB	2GB	100	✓	✓	✓	✓	✗	✓	✓	✗
	osCommerce Web Hosting Pack	N/A	£24	5GB	2GB	100	✓	✓	✓	✓	✗	✓	✓	✗
	ZenCart Web Hosting Pack	N/A	£24	5GB	2GB	100	✓	✓	✓	✓	✗	✓	✓	✗
	PrestaShop Web Hosting Pack	N/A	£24	5GB	2GB	100	✓	✓	✓	✓	✗	✓	✓	✗
123-reg (www.123-reg.co.uk)	Starter	0845 859 0018	£29.88	1GB	5GB	20	✗	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Plus	0845 859 0018	£59.88	5GB	50GB	500	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Pro	0845 859 0018	£107.88	10GB	100GB	750	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Bus Pro	0845 859 0018	£179.88	20GB	Unlimited	1,000	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Plus (MS)	0845 859 0018	£59.88	2GB	25GB	100	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Pro (MS)	0845 859 0018	£107.88	5GB	50GB	500	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Bus Pro (MS)	0845 859 0018	£179.88	10GB	150GB	1,000	✓	✗	✗	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	Light User	0870 321 2020	£45	20MB	1GB	3	✓	✓	✓	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	Everyday	0870 321 2020	£100	200MB	10GB	15	✓	✓	✓	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	Business/Pro	0870 321 2020	£275	500MB	20GB	50	✓	✓	✓	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	JAVA Tomcat	0870 321 2020	£300	100MB	3GB	15	✓	✓	✓	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	ASP.Net	0870 321 2020	£275	100MB	3GB	15	✓	✓	✓	✓	✓	✓	✓	✗
4D Data Centres (www.4dhosting.com)	Bronze Package	0845 166 8386	£40	10MB	500MB	2	✓	✗	✓	✓	✓	✓	✓	✗
4D Data Centres (www.4dhosting.com)	Silver Package	0845 166 8386	£64.99	20MB	2GB	10	✓	✗	✓	✓	✓	✓	✓	✗
4D Data Centres (www.4dhosting.com)	Gold Package	0845 166 8386	£79.99	50MB	2GB	50	✓	✗	✓	✓	✓	✓	✓	✗
4D Data Centres (www.4dhosting.com)	Titanium Package	0845 166 8386	£149.99	500MB	5GB	500	✓	✗	✓	✓	✓	✓	✓	✗
4D Data Centres (www.4dhosting.com)	Reseller Package	0845 166 8386	£299.99	1GB	10GB	1,000	✓	✗	✓	✓	✓	✓	✓	✗
Blackfoot Hosting Ltd (www.blackfoot.co.uk)	Home	N/A	£40	500MB	5GB	5	✓	✓	✓	✓	✓	✓	✓	✗
Blackfoot Hosting Ltd (www.blackfoot.co.uk)	Business	N/A	£50	1GB	20GB	100	✓	✓	✓	✓	✗	✓	✓	✗
Blackfoot Hosting Ltd (www.blackfoot.co.uk)	eCommerce	N/A	£100	2GB	40GB	200	✓	✓	✓	✓	✗	✓	✓	✗
Blackfoot Hosting Ltd (www.blackfoot.co.uk)	Professional	N/A	£150	3GB	60GB	300	✓	✓	✓	✓	✗	✓	✓	✗
Blacknight (www.blacknight.com)	Minimus	35359 918 3072	£45	10MB	200GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
Blacknight (www.blacknight.com)	Medius	35359 918 3072	£79	20GB	400GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
Blacknight (www.blacknight.com)	Maximus	35359 918 3072	£45	30GB	600GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
Bravo14 (http://bravo14.co.uk)	Starter Linux	N/A	£130	2,000MB	2,000MB	10	✓	✓	✓	✓	✓	✓	✓	✓
Bravo14 (http://bravo14.co.uk)	Starter Windows	N/A	£20	2,000MB	2,000MB	10	✓	✓	✓	✓	✓	✓	✓	✓
Bravo14 (http://bravo14.co.uk)	Business Linux	N/A	£45	4,000MB	4,000MB	4,000	✓	✓	✓	✓	✓	✓	✓	✓
Bravo14 (http://bravo14.co.uk)	Business Windows	N/A	£45	4,000MB	4,000MB	4,000	✓	✓	✓	✓	✓	✓	✓	✓
Bravo14 (http://bravo14.co.uk)	Ultimate Linux	N/A	£60	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Bravo14 (http://bravo14.co.uk)	Ultimate Windows	N/A	£60	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Daily Internet (www.daily.co.uk)	Entry	0845 466 2100	£19.28	500MB	5GB	75	✓	✗	✓	✓	✗	✓	✓	✗
Daily Internet (www.daily.co.uk)	Home	0845 466 2100	£50.90	3GB	30GB	30	✓	✓	✓	✓	✗	✓	✓	✗
Daily Internet (www.daily.co.uk)	Business	0845 466 2100	£53.94	12GB	150GB	600	✓	✓	✓	✓	✗	✓	✓	✗
Daily Internet (www.daily.co.uk)	Business Plus	0845 466 2100	£77.94	24GB	250GB	Unlimited	✓	✓	✓	✓	✗	✓	✓	✗
Designwasp (http://designwasp.com)	Starter	0844 372 9848	£30	3GB	20GB	1,000	✓	✗	✓	✗	✗	✓	✗	✗
Designwasp (http://designwasp.com)	Home	0844 372 9848	£40	10GB	50GB	10,000	✓	✗	✓	✗	✗	✓	✓	✗

Get your listing highlighted! Contact Richard

richard.rust@imagine-publishing.co.uk

+44 (0)1202 586436

Featured host of the month: Netcetera www.netcetera.co.uk

Offering more reasons for choosing one of the listed providers

Netcetera was formed in 1996 and since then has gone on to be regarded as one of Europe's premier hosting firms. With a speciality in serving the business sector, it offers universal solutions across three main packages known as Developer, One and Reseller. Each solution is packed with features, with web space and monthly bandwidth providing the tangible differences between the £2.99, £9.99 and £24.99 monthly price points. You can also get a free trial option for one month before switching to the Developer deal, which includes a free development tool and access to all the incredible features available in the standard hosting packages.

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER YEAR	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
Designwasp (http://designwasp.com)	Business	0844 372 9848	£60	Unlimited	Unlimited	Unlimited	✓	✓	✗	✓	✗	✓	✓	✗
Designwasp (http://designwasp.com)	Windows Unlimited	0844 372 9848	£60	Unlimited	Unlimited	Unlimited	✓	✓	✗	✓	✗	✓	✓	✗
Designwasp (http://designwasp.com)	Windows Home	0844 372 9848	£40	10GB	50GB	1,000	✓	✓	✗	✓	✗	✓	✓	✗
Designwasp (http://designwasp.com)	CheapHost	0844 372 9848	£10	300MB	100MB	5	✗	✗	✗	✓	✗	✗	✓	✗
Digital Gibbon Ltd (http://digitalgibbon.com)	Personal	01865 589 990	£12	1GB	Unlimited	10	✓	✗	✗	✓	✗	✓	✓	✗
Digital Gibbon Ltd (http://digitalgibbon.com)	Personal Plus	01865 589 990	£48	Unlimited	Unlimited	50	✓	✗	✗	✓	✗	✓	✓	✗
Digital Gibbon Ltd (http://digitalgibbon.com)	Business	01865 589 990	£108	5GB	Unlimited	1000	✓	✗	✗	✓	✗	✓	✓	✗
Digital Gibbon Ltd (http://digitalgibbon.com)	Business Professional	01865 589 990	£132	Unlimited	Unlimited	Unlimited	✓	✗	✗	✓	✗	✓	✓	✗
Domaincheck (www.domaincheck.co.uk)	Bronze Linux	0191 261 2252	£30	100MB	1GB	5	✗	✓	✓	✓	✓	✓	✓	✓
Domaincheck (www.domaincheck.co.uk)	Gold Linux	0191 261 2252	£100	500MB	5GB	25	✓	✓	✓	✓	✓	✓	✓	✓
Domaincheck (www.domaincheck.co.uk)	Bronze Windows	0191 261 2252	£30	100MB	1GB	5	✗	✓	✓	✓	✓	✓	✓	✓
Domaincheck (www.domaincheck.co.uk)	Gold Windows	0191 261 2252	£100	500MB	5GB	25	✓	✓	✓	✓	✓	✓	✓	✓
Donhost (www.donhost.co.uk)	Reseller Unix	0845 226 5566	£399.99	Unlimited	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✗
Donhost (www.donhost.co.uk)	Reseller Windows	0845 226 5566	£499.99	Unlimited	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✗
Donhost (www.donhost.co.uk)	Enterprise	0845 226 5566	£89.99	1GB	2GB	250	✓	✗	✓	✓	✓	✓	✓	✗
Donhost (www.donhost.co.uk)	Commerce	0845 226 5566	£179.99	2GB	5GB	500	✓	✗	✓	✓	✓	✓	✓	✗
Donhost (www.donhost.co.uk)	Designer	0845 226 5566	£119.99	1GB	2GB	250	✓	✗	✓	✓	✓	✓	✓	✗
Donhost (www.donhost.co.uk)	Developer	0845 226 5566	£259.99	2GB	5GB	500	✓	✗	✓	✓	✓	✓	✓	✗
eHosting (www.ehosting.com)	Starter	0844 999 4100	£23.88	1GB	25GB	10	✗	✗	✗	✗	✗	✗	✓	✓
eHosting (www.ehosting.com)	Personal	0844 999 4100	£59.88	2.5GB	Unlimited	50	✓	✗	✗	✗	✗	✗	✓	✓
eHosting (www.ehosting.com)	Expert	0844 999 4100	£95.88	5GB	Unlimited	250	✓	✗	✗	✗	✗	✗	✓	✓
eHosting (www.ehosting.com)	Virtual	0844 999 4100	£227.88	50GB	Unlimited	Unlimited	✓	✗	✗	✓	✗	✓	✓	✓
Equiphase (www.equiphase.net)	Bronze	0121 314 4865	£30	200MB	2GB	10	✓	✓	✓	✓	✗	✓	✗	✓
Equiphase (www.equiphase.net)	Silver	0121 314 4865	£42	400MB	5GB	20	✓	✓	✓	✓	✗	✓	✗	✓
Equiphase (www.equiphase.net)	Gold	0121 314 4865	£72	800MB	10GB	100	✓	✓	✓	✓	✗	✓	✗	✓
Equiphase (www.equiphase.net)	Platinum	0121 314 4865	£114	1,200MB	40GB	200	✓	✓	✓	✓	✗	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Email Only	02380 249 823	£40	1GB	2GB	10	✗	✗	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Essential	02380 249 823	£75	2GB	5GB	10	✗	✗	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Superior	02380 249 823	£140	5GB	10GB	25	✓	✓	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Premium	02380 249 823	£250	10GB	25GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Starter	N/A	£29.99	500MB	1GB	3	✓	✓	✓	✓	✗	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Home	N/A	£54.99	2.5GB	30GB	50	✓	✓	✓	✓	✗	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Business	N/A	£79.99	6.5GB	Unlimited	Unlimited	✓	✓	✓	✓	✗	✓	✓	✓
Evohosting (www.evohosting.co.uk)	eCommerce	N/A	£159.99	30GB	Unlimited	Unlimited	✓	✓	✓	✓	✗	✓	✓	✓
Fasthosts www.fasthosts.co.uk	Personal Standard	0844 583 0777	£53.88	5GB	Unlimited	500	✓	✗	✓	✓	✓	✓	✓	✗
	Business Standard	0844 583 0777	£95.88	25GB	Unlimited	1,000	✓	Option	✓	✓	✓	✓	✓	✗
	Business Premium	0844 583 0777	£173.88	50GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
	WD Starter Reseller	0844 583 0777	£149.99	20GB	Unlimited	Unlimited	✓	Option	Option	✓	✓	✓	✓	✗
	Advanced Reseller	0844 583 0777	£199.99	Unlimited	Unlimited	Unlimited	✓	Option	Option	✓	✓	✓	✓	✗
Giacom (www.giacom.com)	Business Pro	0800 542 7500	£199	100MB	2GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Starter Professional	0845 644 7750	£29.80	2.5GB	10GB	1,000	✗	✗	✗	✓	✗	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Home Professional	0845 644 7750	£89.99	10GB	50GB	10,000	✓	✗	✗	✓	✗	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Business Professional	0845 644 7750	£129.99	Unlimited	Unlimited	Unlimited	✓	✗	✗	✓	✗	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Reseller Professional	0845 644 7750	£299.99	Unlimited	Unlimited	Unlimited	✓	✗	✗	✓	✗	✓	✓	✓
Hostway (www.hostway.co.uk)	Silver	0808 180 1880	£79.50	150MG	3GB	5	✗	Option	✓	✗	✓	✓	✓	✗
Hostway (www.hostway.co.uk)	Gold	0808 180 1880	£139.50	300MB	5GB	10	✓	Option	✓	✗	✓	✓	✓	✗
Hostway (www.hostway.co.uk)	Gold Plus	0808 180 1880	£189.50	450MB	10GB	30	✓	Option	✓	✗	✓	✓	✓	✗
Hostway (www.hostway.co.uk)	Platinum	0808 180 1880	£359.50	600MB	20GB	50	✓	Option	✓	✗	✓	✓	✓	✗
Hostway (www.hostway.co.uk)	Platinum Plus	0808 180 1880	£599.50	1.2GB	40GB	10	✓	Option	✓	✗	✓	✓	✓	✗
Hostway (www.hostway.co.uk)	Email Plus	0808 180 1880	£49.95	50MB	N/A	5	N/A	N/A	N/A	✗	✓	✓	✓	✗
ICUK www.icukhosting.co.uk	Professional	0845 009 9175	£30	250MB	1GB	50	✓	✓	option	✓	✓	✓	✓	✗
ICUK www.icukhosting.co.uk	Advanced	0845 009 9175	£50	2GB	2.5GB	150	✓	✓	option	✓	✓	✓	✓	✗
ICUK www.icukhosting.co.uk	Enterprise	0845 009 9175	£80	2GB	500MB	50	✓	✓	option	✓	✓	✓	✓	✗
ICUK www.icukhosting.co.uk	Professional Plus	0845 009 9175	£90	500MB	5GB	100	✓	✓	option	✓	✓	✓	✓	✗

Hosting listings

Our web-hosting guide is brought to you in association with Fasthosts, the UK's leading reseller web-hosting provider
- **Unlimited** websites
- **Unlimited** bandwidth
- **Unlimited** web space
- Use your own brand throughout
- 24/7 expert UK-based support
- No-risk trial – 3 months free

Keep an eye on the latest packages and deals with our comprehensive list of service providers

Discover what an enhanced listing can do for your business. Contact Richard on:
+44 (0)1202 586436

Continued...

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER YEAR	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
ICUK www.icukhosting.co.uk	Premium Plus	0845 009 9175	£150	1GB	12.5GB	500	✓	✓	option	✓	✓	✓	✓	✗
ICUK www.icukhosting.co.uk	Enterprise Plus	0845 009 9175	£300	2GB	20GB	Unlimited	✓	✓	option	✓	✓	✓	✓	✗
ICUK www.icukhosting.co.uk	Reseller Windows	0845 009 9175	£200	Unlimited	Unlimited	Unlimited	✓	✓	option	✓	✓	✓	✓	✗
ICUK www.icukhosting.co.uk	Reseller Linux	0845 009 9175	£200	Unlimited	Unlimited	Unlimited	✓	✓	option	✓	✓	✓	✓	✗
ICUK www.icukhosting.co.uk	Reseller Enterprise	0845 009 9175	£500	Unlimited	Unlimited	Unlimited	✓	✓	option	✓	✓	✓	✓	✗
JAB Web Hosting (www.jabwebhosting.com)	Mail - M105	0800 043 0153	£15.17	256MB	3GB	50	✗	✗	✗	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Home - H25	0800 043 0153	£24.47	1GB	8GB	50	✓	✗	✗	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Professional - P55	0800 043 0153	£45	2GB	15GB	75	✓	✓	✓	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Reseller - R25	0800 043 0153	£97.88	2GB	30GB	250	✓	✗	✓	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Reseller - R105	0800 043 0153	£271.60	4GB	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Blog	01438 342 490	£20	100MB	1GB	5	✗	✗	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Starter	01438 342 490	£30	1GB	1GB	10	✗	✗	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Starter	01438 342 490	£30	1GB	1GB	10	✗	✗	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Dynamic	01438 342 490	£50	2GB	2GB	20	✓	✓	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Premium	01438 342 490	£80	5GB	5GB	50	✓	✓	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Unlimited	01438 342 490	£120	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
LD Hosts (http://ldhosts.co.uk)	LD Budget (Linux)	07891 235858	£11.88	1GB	1GB	500	✓	✓	✗	✓	✓	✓	✓	✗
LD Hosts (http://ldhosts.co.uk)	LD Home (Linux)	07891 235858	£23.88	10GB	10GB	5,000	✓	✓	✗	✓	✓	✓	✓	✗
LD Hosts (http://ldhosts.co.uk)	LD Pro (Linux)	07891 235858	£41.88	50GB	50GB	20,000	✓	✓	✗	✓	✓	✓	✓	✗
LD Hosts (http://ldhosts.co.uk)	LD Unlimited (Linux)	07891 235858	£60	Unlimited	Unlimited	Unlimited	✓	✓	✗	✓	✓	✓	✓	✗
LD Hosts (http://ldhosts.co.uk)	Windows Home	07891 235858	£17.88	2GB	2GB	1,000	✓	✓	✗	✓	✓	✓	✓	✗
LD Hosts (http://ldhosts.co.uk)	Windows Unlimited	07891 235858	£60	Unlimited	Unlimited	Unlimited	✓	✓	✗	✓	✓	✓	✓	✗
Media Temple (http://mediatemple.net)	Shared-Server Pro	+1 310 841 5500	£64	2GB	1TB	1,000	✓	✓	✓	✓	✓	✓	✓	✓
Media Temple (http://mediatemple.net)	Shared-Server Advanced	+1 310 841 5500	£109	5GB	1.5TB	5,000	✓	✓	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Starter	01604 212904	£37.08	10GB	100GB	Unlimited	✗	✗	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Home Pro	01604 212904	£62.28	25GB	500GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Business	01604 212904	£103.08	75GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Reseller	01604 212904	£103.08	100GB	Unlimited	No-limit	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	StartUp Plus	0845 363 3632	£89.99	500MB	5GB	10	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	Business	0845 363 3632	£149.99	2,000MB	20GB	100	✗	✗	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	Business Plus	0845 363 3632	£89.99	3,000MB	30GB	200	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	Designer	0845 363 3632	£239.88	1GB	20GB	200	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	Designer Plus	0845 363 3632	£479.88	Unlimited	30GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
NETCETERA Netcetera www.netcetera.co.uk	DEVELOPER	0800 061 2801	£32.89	1GB	Unlimited	500	✓	✓	✓	✓	✓	✓	✓	✓
	ONE	0800 061 2801	£109.99	5GB	Unlimited	1000	✓	✓	✓	✓	✓	✓	✓	✓
	RESELLER	0800 061 2801	£274.89	Unlimited	Unlimited	1000	✓	✓	✓	✓	✓	✓	✓	✓
	VM500 Server	0800 061 2801	£300	20GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
	2200DC Server	0800 061 2801	£720	160GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
	3000DC Server	0800 061 2801	£1,200	2x500GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
	2600QC Server	0800 061 2801	£1,800	2x500GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	Shared 100	0207 1000 424	£60	100MB	1GB	5	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	VS100	0207 1000 424	£600	5GB	5GB	5	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	VS200	0207 1000 424	£1,000	10GB	10GB	10	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	VS300	0207 1000 424	£2,000	15GB	50GB	20	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	Dedicated Servers	0207 1000 424	£3,000+	73GB+	1,500GB	100+	✓	✓	✓	✓	✓	✓	✓	✓
PurplePaw (www.purplepaw.co.uk)	Email	N/A	£25	100MB	1GB	10	✗	✗	✓	✓	✗	✓	✗	✗
PurplePaw (www.purplepaw.co.uk)	Play	N/A	£40	100MB	1GB	10	✗	✗	✓	✓	✗	✓	✗	✗
PurplePaw (www.purplepaw.co.uk)	Plus	N/A	£65	750MB	5GB	25	✓	✓	✓	✓	✗	✓	✓	✗
PurplePaw (www.purplepaw.co.uk)	Power	N/A	£95	2GB	10GB	100	✓	✓	✓	✓	✗	✓	✓	✗
PurplePaw (www.purplepaw.co.uk)	R3 Reseller	N/A	£660	10GB	50GB	Unlimited	✓	✓	✓	✓	✗	✓	✓	✗
Reddex UK (www.reddexuk.com)	Reddex Design Starter	0843 289 4625	£18	1GB	Unlimited	10	✓	✓	✓	✓	✓	✓	✓	✓
Reddex UK (www.reddexuk.com)	Reddex Design Business	0843 289 4625	£59.88	10GB	Unlimited	100	✓	✓	✓	✓	✓	✓	✓	✓
Reddex UK (www.reddexuk.com)	Reddex Design Premium	0843 289 4625	£107.88	100GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Skymarket (www.skymarket.co.uk)	Standard 1	0800 321 7788	£49	10MB	2GB	1	✓	✗	✓	✓	✓	✓	✓	✓
Skymarket (www.skymarket.co.uk)	Standard 2	0800 321 7788	£69	20MB	2GB	1	✓	✗	✓	✓	✓	✓	✓	✓

Get your listing highlighted! Contact Richard
richard.rust@imagine-publishing.co.uk
+44 (0)1202 586436

Can you recommend your host?

Tweet us today with your hosting comments and advice
@WebDesignerMag

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER YEAR	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
Skymarket (www.skymarket.co.uk)	Premium 1	0800 321 7788	£99	25MB	2GB	1	✓	✗	✓	✓	✓	✓	✓	✓
Skymarket (www.skymarket.co.uk)	Premium 2	0800 321 7788	£149	50MB	2GB	1	✓	✗	✓	✓	✓	✓	✓	✓
STRATO Hosting (www.strato-hosting.co.uk)	STRATO BasicWeb	00800 8007 0070	£48	2GB	Unlimited	1,000	✗	✗	✓	✓	✓	✓	✓	✗
STRATO Hosting (www.strato-hosting.co.uk)	STRATO PowerWeb	00800 8007 0070	£72	6GB	Unlimited	2,000	✓	✗	✓	✓	✓	✓	✓	✗
STRATO Hosting (www.strato-hosting.co.uk)	STRATO AdvancedWeb	00800 8007 0070	£120	20GB	Unlimited	4,000	✓	✗	✓	✓	✓	✓	✓	✗
STRATO Hosting (www.strato-hosting.co.uk)	STRATO EnterpriseWeb	00800 8007 0070	£180	50GB	Unlimited	6,000	✓	✗	✓	✓	✓	✓	✓	✗
Streamline.net (www.streamline.net)	Trial (3 month)	0844 941 1000	N/A	10GB	Unlimited	1,000	✓	Option	✓	✓	✓	✓	✓	✗
Streamline.net (www.streamline.net)	Starter	0844 941 1000	£23.88	500MB	Unlimited	20	✓	Option	✓	✓	✓	✓	✓	✗
Streamline.net (www.streamline.net)	Personal	0844 941 1000	£41.88	3GB	Unlimited	500	✓	Option	✓	✓	✓	✓	✓	✗
Streamline.net (www.streamline.net)	Plus	0844 941 1000	£71.88	10GB	Unlimited	1,000	✓	Option	✓	✓	✓	✓	✓	✗
Streamline.net (www.streamline.net)	Multisite	0844 941 1000	£137.88	20GB	Unlimited	Unlimited	✓	Option	✓	✓	✓	✓	✓	✗
Swish Hosting (www.swishhosting.co.uk)	Email	08445 67 69 71	£18	-	Unlimited	Unlimited	✗	✗	✓	✓	✓	✓	✓	✓
Swish Hosting (www.swishhosting.co.uk)	Windows Hosting	08445 67 69 71	£66	-	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✓
Swish Hosting (www.swishhosting.co.uk)	Linux Hosting	08445 67 69 71	£66	-	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✓
Swish Hosting (www.swishhosting.co.uk)	eCommerce	08445 67 69 71	£90	-	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Swish Hosting (www.swishhosting.co.uk)	SiteBuilder	08445 67 69 71	£12	-	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Switch Media (www.switchmedia.com)	Switch Standard	0151 236 9111	£159	1GB	50GB	500	✓	✗	✗	✗	✓	✓	✗	✗
Switch Media (www.switchmedia.com)	Switch Business	0151 236 9111	£249	10GB	150GB	1500	✓	✗	✗	✗	✓	✓	✗	✗
Switch Media (www.switchmedia.com)	Business Pro	0151 236 9111	£348.96	20GB	300GB	5000	✓	✗	✗	✗	✓	✓	✗	✗
thename.co.uk (www.thename.co.uk)	Parking	0870 765 6364	£52.88	25MB	500MB	15	✓	✓	✓	✓	✓	✓	✓	✓
thename.co.uk (www.thename.co.uk)	Forwarding	0870 765 6364	From £15	N/A	500MB	N/A	✗	✗	✓	✗	✓	✓	✓	✗
thename.co.uk (www.thename.co.uk)	Hosting	0870 765 6364	From £7.50	N/A	N/A	N/A	✗	✗	✓	✗	✓	✓	✓	✗
TwentyHost (www.twentyhost.co.uk)	Basic5S	0845 641 0776	£24	100MB	1,500MB	25	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Standard5S	0845 641 0776	£45	200MB	3,000MB	50	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Business5S	0845 641 0776	£70	500MB	7,500MB	100	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Advanced5S	0845 641 0776	£110	1,000MB	15,000MB	200	✓	✓	✓	✓	✓	✓	✓	✓
VARiHOST (www.varihost.net)	WordPress Basic	0208 144 7057	£47.88	2GB	10GB	10	✓	✗	✓	✓	✓	✓	✓	✗
VARiHOST (www.varihost.net)	WordPress Plus	0208 144 7057	£71.88	4GB	100GB	100	✓	✗	✓	✗	✓	✓	✓	✗
VARiHOST (www.varihost.net)	WordPress Extra	0208 144 7057	£119.88	unlimited	Unlimited	Unlimited	✓	✗	✓	✗	✓	✓	✓	✗
WebFusion (www.webfusion.co.uk)	Fusion Professional	0845 130 1602	£107.40	5GB	50GB	1,000	✓	✗	✓	✓	✓	✓	✓	✗
WebFusion (www.webfusion.co.uk)	Fusion Business	0845 130 1602	£179.40	10GB	150GB	1,500	✓	✗	✓	✓	✓	✓	✓	✗
WebFusion (www.webfusion.co.uk)	Fusion Developer	0845 130 1602	£227.40	20GB	300GB	5,000	✓	✗	✓	✓	✓	✓	✓	✗
WebFusion (www.webfusion.co.uk)	Fusion Reseller	0845 130 1602	£329.99	Unlimited	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✗
Web Wiz (www.webwiz.co.uk)	Windows Starter	0844 358 1450	£69.95	1GB	25GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Web Wiz (www.webwiz.co.uk)	Windows Professional	0844 358 1450	£149.95	4GB	100GB	500	✓	✓	✓	✓	✓	✓	✓	✓
Web Wiz (www.webwiz.co.uk)	Windows Starter Reseller	0844 358 1450	£199.95	5GB	25GB	500	✓	✓	✓	✓	✓	✓	✓	✓
Web Wiz (www.webwiz.co.uk)	Windows Business Reseller	0844 358 1450	£499.95	20GB	200GB	5,000	✓	✓	✓	✓	✓	✓	✓	✓
Zebra Internet (www.zebrainternet.com)	Starter Pro	N/A	£10.99	5GB	40GB	50	✓	✓	✓	✓	✓	✗	✓	✓
Zebra Internet (www.zebrainternet.com)	Business Pro	N/A	£46.99	10GB	80GB	500	✓	✓	✓	✓	✓	✗	✓	✓
Zebra Internet (www.zebrainternet.com)	Ultimate Pro	N/A	£79.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✗	✓	✓
Zen Internet www.zen.co.uk	Bronze (Linux)	0845 058 9000	£47.88	2GB	20GB	10	✓	✓	✓	✓	✓	✓	✓	✗
Zen Internet www.zen.co.uk	Silver (Linux)	0845 058 9000	£95.88	5GB	50GB	25	✓	✓	✓	✓	✓	✓	✓	✗
Zen Internet www.zen.co.uk	Gold (Linux)	0845 058 9000	£143.88	10GB	100GB	50	✓	✓	✓	✓	✓	✓	✓	✗
Zen Internet www.zen.co.uk	Platinum (Linux)	0845 058 9000	£239.88	50GB	250GB	100	✓	✓	✓	✓	✓	✓	✓	✗
Zen Internet www.zen.co.uk	Reseller (Linux)	0845 058 9000	£479.88	150GB	500GB	250	✓	✓	✓	✓	✓	✓	✓	✗
Zen Internet www.zen.co.uk	Designer (Windows)	0845 058 9000	£59.88	2GB	20GB	-	✓	✗	✗	✓	✓	✓	✓	✗
Zen Internet www.zen.co.uk	Developer (Windows)	0845 058 9000	£179.88	10GB	100GB	-	✓	✗	✗	✓	✓	✓	✓	✗

Golden rules to top hosting

We identify and explain the key criteria for success…

1 The best resources for you

Selecting your ideal package is largely determined by the kinds of resources and quantity of features you require from your hosting solution. Key criteria like web space and monthly bandwidth are important for those services likely to be subjected to heavy amounts of traffic, which is why package solutions for enterprise applications are typically much more expensive. The general rule of thumb is not to buy more than you need or underestimate potential requirements…

2 Competitive and reliable

The hosting market is big business and hosting providers do try to drive prices down in a bid to entice your custom. Use our chart to compare costs, but be sure to visit the vendor websites to keep track of the latest deals, as they change. Remember that low price should not always be a deciding factor and that paying a premium for a more reliable, trusted and experienced vendor can offer you much better value for money in the long term.

3 Putting you in control

Modern hosting is all about giving customers the power to set up, monitor and maintain their web space with minimal fuss. Most commercial vendors offer access to award-winning and intuitive control panels that enable you to log in remotely and intuitively tweak your account, without the need to relay complicated instructions down the phone. Be sure to find out from your potential host as much as you can about the control panel and request a demo.

4 Fantastic customer support

If all else fails and you need some extra help to get your hosting back online, then a commitment to future customer support is key. Many vendors offer a service-level agreement which outlines what you can expect here, however most will be more explicit about whether phone support is included or email contact is preferred. Think about what you need for peace of mind and factor good, comprehensive technical support against the price.

Your favourite magazines n
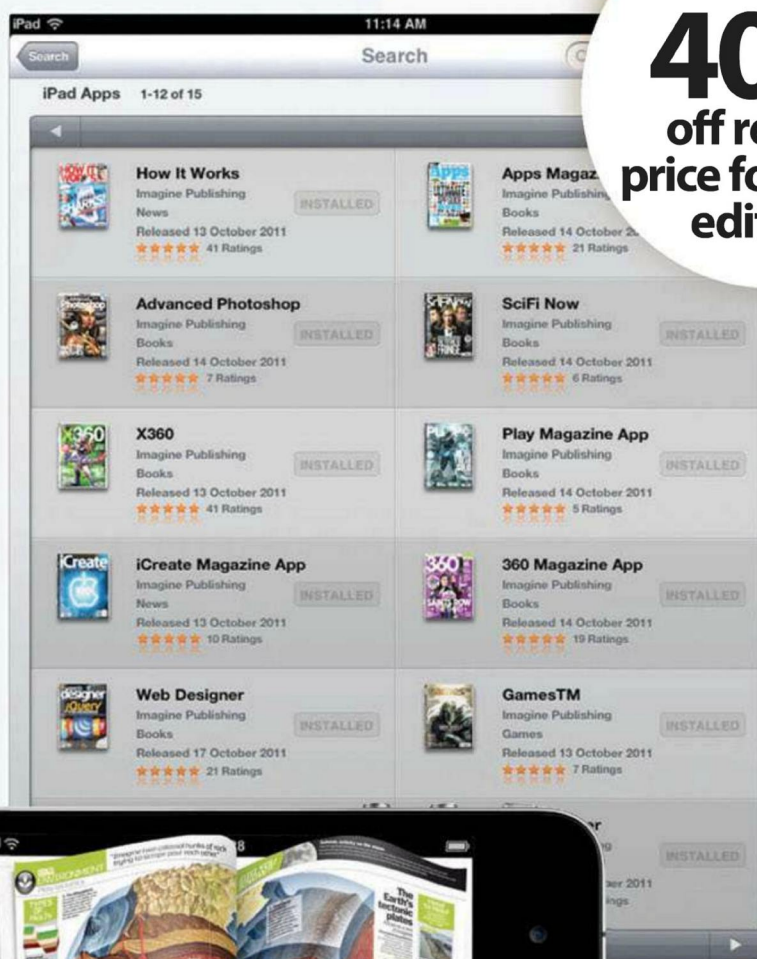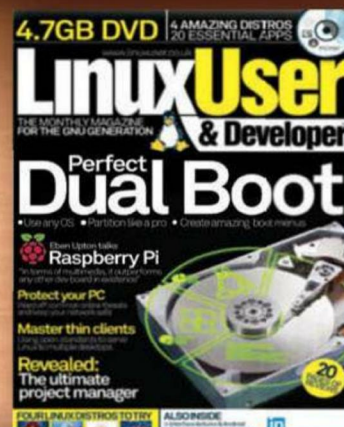
3D Artist
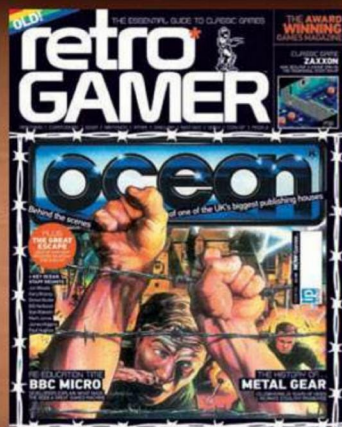
Retro Gamer

X360

SciFiNow

How It Works

Linux User & Developer

Available now on the Apple iOS 5 Newsstand app
Search for 'Imagine

Your free resource disc
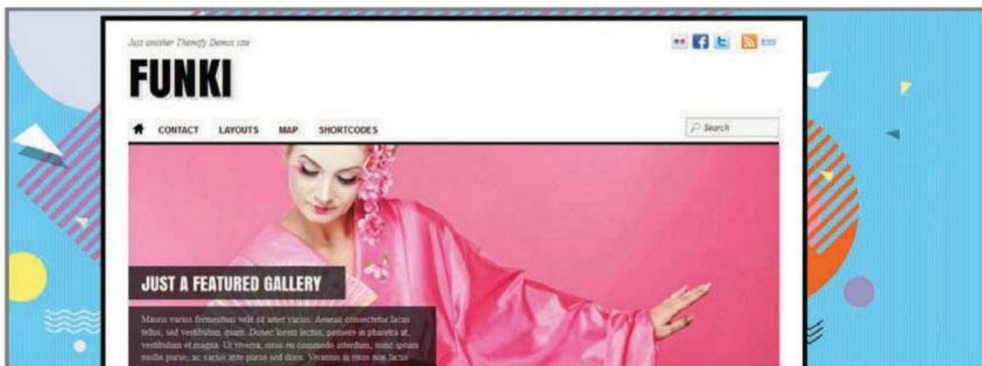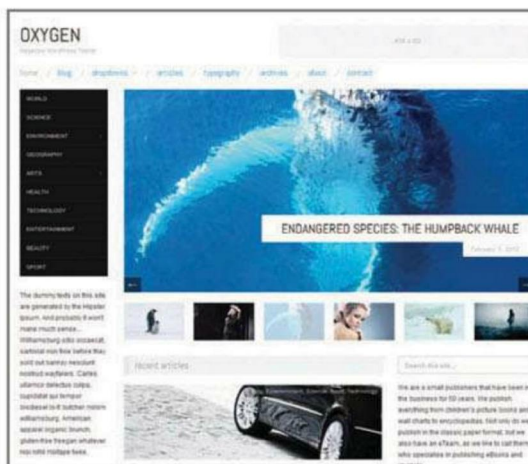
This disc is worth more than **£100**

Welcome to the Web Designer cover disc, a feature-packed CD containing a must-have collection of resources and tutorials

Once again **Web Designer** has scoured the world wide web and beyond searching for the best selection of designer resources out there. The latest lineup includes two fully responsive WordPress themes, 15 hi-res tile textures, 20 stitch circle vectors, and (for a bit of fun) the Shiver Me Timbers pirate font – yarrr!

24
inspiring backdrops
This issue's Design Cloud offers 24 sites with big, beautiful full-screen backgrounds

7
tutorial files
All the code and assets needed to complete this month's workshops

4
stunning blogs
Blog Beautiful brings together four of the best blogs from some of the web's leading creative agencies

Two responsive WordPress themes

From: **themify.me / alienwp.com**

This issue **Web Designer** brings you two top-quality and contemporary responsive themes. The Funki theme is coded with HTML5 techniques and has seven layouts and three sidebar options. Oxygen is a neat, minimalist magazine theme with a featured image slider. Both templates are the perfect solution for blogs on mobile and desktop.

Load the CD:
Microsoft Windows
We support Windows 7 unless otherwise stated. The CD-ROM should autorun once placed into your disc drive. If not, follow these instructions:

1 Browse to My Computer.

2 Right-click on your CD drive, and select Open from the drop-down list.

3 Read the 'readme.txt' if there is one present to find out which files you need to launch to run the interface.

Load the CD:
Mac OS X
Up to OS X 10.7 (Lion) supported. This CD-ROM interface will NOT autorun when placed into your CD drive. Instead:

1 Double-click the CD icon on your desktop.

2 Read the 'readme' file if there is one present to find out which file you require to successfully run the interface.

What would you like to see featured on your free CD? webdesigner@imagine-publishing.co.uk

20 stitch-like circle vectors
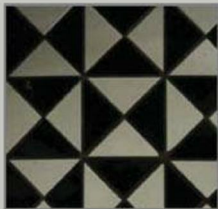
From: designfruit.com

This collection of vectors is inspired by a host of quilt designs, circular borders and intricate, complex circle patterns, as if they were stitched or sewn. This image set combines lacy filigree with dot patterns, zigzag lines, geometric shapes and ornate, Victorian embellishments.

Shiver Me Timbers font

From: www.nicksfonts.com

This is a pirate-themed font, based loosely on several of Victor Hammer's Uncial typefaces, designed between 1925 and 1953. It boasts a weathered and corroded look for authenticity. Alongside the font the bullet character acts as an 'X marks the spot' while the section mark provides the true pirate symbol – a skull and crossbones.

15 tile textures

From: arsenal. gomedia.us

This collection contains 15 high-resolution, real-life tile textures, providing a host of repeating tiling patterns with bold, geometric and mosaic designs. These are great textures for page backgrounds with a file dimension of 3,264 x 2,448px at 180dpi.

30 music loops

Web Designer's friends at Soundsnap (www. soundsnap.com) once again provide an essential mini library of music and FX clips to add an extra audio dimension to any webpage.

web designer 195

- 2 responsive WordPress themes
- 15 repeating tile textures
- 20 Stitchy circle vectors
- Shiver Me Timbers font
- 30 Soundsnap audio loops
- 110 minutes of video
- Tutorial files

Web Designer © Imagine Publishing 2012

Disc problems

If the menu doesn't run or your disc is faulty, we will replace it free of charge. Drop an email listing the problem to WDxtrahelp@imagine-publishing. co.uk. Please don't send mail about the disc to the editorial team. If you are having problems getting the best out of programs or extras provided, then please contact the relevant vendor.

Over 100 minutes of HTML5, CSS3 and ActionScript video tutorials
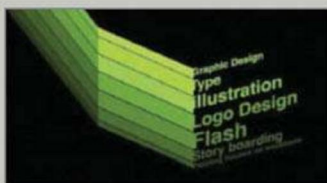
AS3 Menus

The fifth and final instalment of this five-part series covers keyboard-controlled menus and how to go about neatly wrapping up a project.

From: www.cartoonsmart.com

KILLERSITES.COM
LEARN TO BUILD 'KILLER' WEB SITES.

HTML5, CSS3 & JQuery
Crossbrowser Compatability
by Stefan Romano

HTML5, CSS3 & jQuery

The latest trio of videos from KillerSites takes a look at the logistics behind cross-browser compatibility, HTML5 layouts and HTML boilerplates.

From: www.killersites.com

Find more free tutorial content at www.webdesignermag.co.uk

Fonts

Back at the turn of the century fonts were still styled inline. A great example is found hidden in the source code of the old BBC website:

```
001 <font size="1">
002 <span
class="date">Saturday,
16 June, 2001, 04:49
GMT 05:49 UK</span>
003 </font>
```


2001
Web design relic

Tables

Tables were the layout king before CSS made its entrance. The 2001 version of the BBC site uses a mixture of tables to position the majority of the content while employing classes to add body text.

Load time!

Page size: **141 KB**

09 seconds on a 128 kbps connection

2008
Web design relic

Widescreen

The 2008 version of BBC Sport was the precursor to the 2012 makeover. It still boasted the three-column layout of previous iterations but went widescreen for use with more contemporary screen resolutions.

BBC Sport www.bbc.co.uk/sport

The BBC Sport site has been a long-time bastion of sports coverage and it has gone through a number of stages in its evolution

The BBC Sport website is one of the most visited sports sites in the world and not without good reason. It boasts comprehensive and current coverage of some of the most popular sporting events across the globe. The site has always been predominantly about the content, while the design has been about offering the best user experience.

Back in 2001 the site provided a breadth of sports coverage that it still retains to this day, but technology restricted its design ethos. The site structure was a simple three-column layout with navigation left, content middle and features right – a stance that remained until recently; all this typically retained inside a skinny 600px-wide wrapper.

The site structure was a simple three-column layout with navigation left, content middle and features right